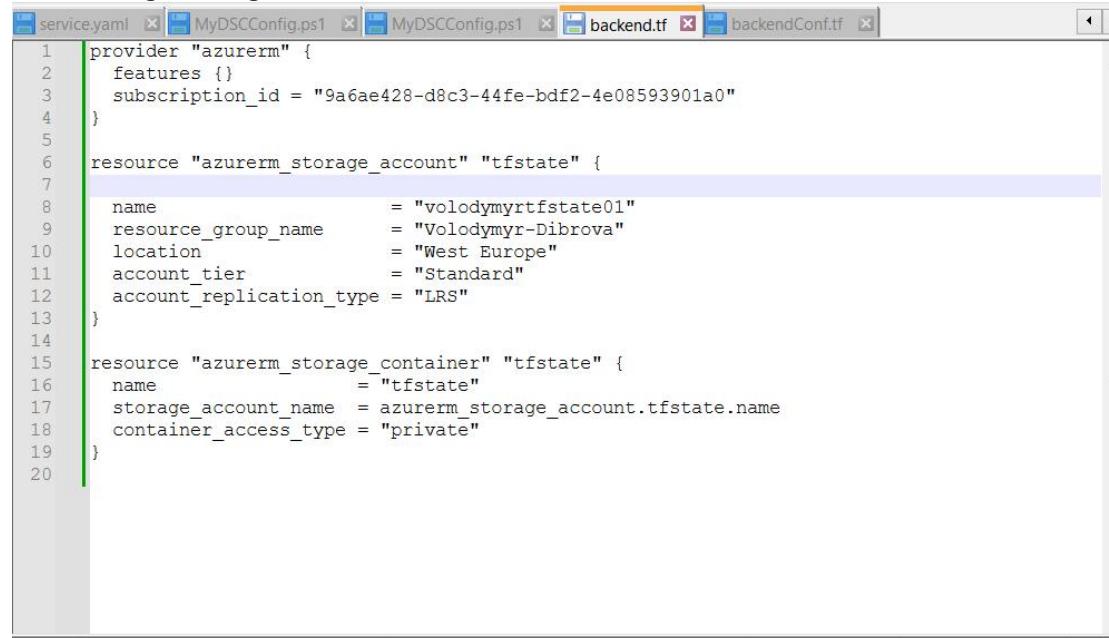


Practical Task 1: Install, Configure, and Manage Terraform State in Azure Requirements:

- Install Terraform on your local machine.
- Verify the installation by checking the Terraform version.
- Authenticate with Azure using `az login` and configure Terraform for Azure authentication.
- Create a **Terraform backend** configuration using an **Azure Storage Account** to store the Terraform state remotely:
 - o Define a storage account, a container, and a blob in **Terraform configuration**.
 - o Use `terraform init` to initialize the backend.
 - o Run `terraform apply` to deploy the storage account for state management.
 - o Verify that the Terraform state file is stored in the Azure Storage Account.
 - o Implement basic **state locking** using Azure blob storage.
 - o Destroy the storage account (after confirming the state behavior).

1. Creating Storage Account and container



```
service.yaml MyDSCConfig.ps1 MyDSCConfig.ps1 backend.tf backendConf.tf
1 provider "azurerm" {
2     features {}
3     subscription_id = "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
4 }
5
6 resource "azurerm_storage_account" "tfstate" {
7     name          = "volodymyrftfstate01"
8     resource_group_name = "Volodymyr-Dibrova"
9     location      = "West Europe"
10    account_tier   = "Standard"
11    account_replication_type = "LRS"
12 }
13
14 resource "azurerm_storage_container" "tfstate" {
15     name          = "tfstate"
16     storage_account_name = azurerm_storage_account.tfstate.name
17     container_access_type = "private"
18 }
19
20 }
```

```

Windows PowerShell
PS E:\projects\terraformProj\terraform-azure-state> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/azurerm from the dependency lock file
- Using previously-installed hashicorp/azurerm v4.18.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS E:\projects\terraformProj\terraform-azure-state> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create

Terraform will perform the following actions:

# azurerm_storage_account.tfstate will be created
+ resource "azurerm_storage_account" "tfstate" {
  + access_tier           = (known after apply)
  + account_kind          = "StorageV2"
  + account_replication_type = "LRS"
  + account_tier          = "Standard"
  + allow_nested_items_to_be_public = true
}

```

Created blob file

service.yaml MyDSConfig.ps1 MyDSConfig.ps1 backend.tf backendConf.tf

```

1 terraform {
2   backend "azurerm" {
3     resource_group_name = "Volodymyr-Dibrova"
4     storage_account_name = "volodymyrtfstate01"
5     container_name       = "tfstate"
6     key                  = "terraform.tfstate"
7   }
8 }

```

Microsoft Azure

Home > Volodymyr-Dibrova > volodymyrtfstate01 | Containers >

tfstate Container

Upload Change access level Refresh Delete Change tier Acquire lease Break lease View snapshots Create snapshot Give feedback

Search resources, services, and docs (G+) Copilot

Authentication method: Access key (Switch to Microsoft Entra user account)

Location: tfstate

Search blobs by prefix (case-sensitive): Show deleted blobs

Add filter

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state	...
terraform.tfstate	2/10/2025 9:19:04 PM	Hot (inferred)	Not yet archived	Block blob	10.74 KB	Available	...

```

PS E:\projects\terraformProj\terraform-azure-state> az storage blob list \
--account-name volodymyrtfstate01 \
--container-name tfstate \
--output table

There are no credentials provided in your command and environment, we will query for account key for your storage account.
It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials.

You also can add '--auth-mode login' in your command to use Azure Active Directory (Azure AD) for authorization if your login account is assigned required RBAC roles.
For more information about RBAC roles in storage, visit https://learn.microsoft.com/azure/storage/common/storage-auth-ad-rbac-cli.

In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get more information about environment variable usage.
Name      Blob Type  Blob Tier  Length  Content Type    Last Modified      Snapshot
-----  -----  -----  -----  -----  -----  -----
terraform.tfstate  BlockBlob  Hot  10995  application/json  2025-02-10T19:19:04+00:00

```

Practical Task 2: Deploy an Azure Virtual Machine with a Custom Network and Security Rules

Requirements:

Extend the Terraform configuration to deploy:

- An Azure Virtual Network (VNet) with a **custom subnet**.
- A Network Security Group (NSG) with the following rules:
 - Allow SSH (port 22) inbound for a specific IP range.
 - Allow HTTP (port 80) inbound for all users.
 - Deny all other inbound traffic.
- A Public IP Address assigned to the VM.
- An Azure Virtual Machine (VM) using an Ubuntu image, attached to the subnet and NSG.
- A Terraform output variable to display the public IP of the VM after deployment.
- Use Provisioners to run a startup script that installs and starts an Nginx web server on the VM.
- Verify:
 - That SSH access works for the specified IP range.
 - That the Nginx web page is accessible via the VM's public IP.
- Destroy the infrastructure when complete.

backend.tf : Main file

variables.tf : All variables

network.tf : VNet, Subnet, NSG

compute.tf : VM

outputs.tf : Public IP output

backend.tf: Main file

```
File Edit Selection View Go Run Terminal Help ← → ○ task2

EXPLORER ... backend.tf ●
TASK2
  .terraform
  .terraform.lock.hcl
  backend.tf
  compute.tf
  network.tf
  outputs.tf
  terraform.tfstate
  terraform.tfstate.backup
  variables.tf

1  terraform {
2   required_providers {
3     azurerm = {
4       source  = "hashicorp/azurerm"
5       version = ">=3.0.0"
6     }
7   }
8 }
9
10 provider "azurerm" {
11   features {}
12   subscription_id = var.subscription_id
13 }
14
15 data "azurerm_resource_group" "rg" {
16   name = var.resource_group_name
17 }
18
19
```

variables.tf : All variables

File Edit Selection View Go Run Terminal Help ← → task2

EXPLORER ...

TASK2

- > .terraform
- ≡ .terraform.lock.hcl
- variables.tf
- compute.tf
- network.tf
- outputs.tf
- terraform.tfstate
- terraform.tfstate.backup

variables.tf

```

1 variable "subscription_id" {
2   description = "Azure Subscription ID"
3   default     = "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
4 }
5
6 variable "resource_group_name" {
7   description = "Resource Group Name"
8   default     = "Volodymyr-Dibrova"
9 }
10
11 variable "location" {
12   description = "Azure region"
13   default     = "westeurope"
14 }
15
16 variable "vnet_name" {
17   description = "Virtual Network Name"
18   default     = "example-vnet"
19 }
20
21 variable "subnet_name" {
22   description = "Subnet Name"
23   default     = "example-subnet"
24 }
25
26 variable "nsg_name" {
27   description = "Network Security Group Name"
28   default     = "example-nsg"
29 }
30
31 variable "public_ip_name" {
32   description = "Public IP Name"
33   default     = "example-public-ip"
34 }
35
36 variable "vm_name" {
37   description = "Virtual Machine Name"
}

```

OUTLINE

TIMELINE

network.tf : VNet, Subnet, NSG

File Edit Selection View Go Run Terminal Help ← → task2

EXPLORER ...

TASK2

- > .terraform
- ≡ .terraform.lock.hcl
- variables.tf
- backend.tf
- compute.tf
- network.tf
- outputs.tf
- terraform.tfstate
- terraform.tfstate.backup

network.tf

```

1 resource "azurerm_virtual_network" "vnet" {
2   name          = var.vnet_name
3   location      = data.azurerm_resource_group.rg.location
4   resource_group_name = data.azurerm_resource_group.rg.name
5   address_space    = ["10.0.0.0/16"]
6 }
7
8 resource "azurerm_subnet" "subnet" [
9   name          = var.subnet_name
10  resource_group_name = data.azurerm_resource_group.rg.name
11  virtual_network_name = azurerm_virtual_network.vnet.name
12  address_prefixes  = ["10.0.1.0/24"]
13 ]
14
15 resource "azurerm_network_security_group" "nsg" {
16   name          = var.nsg_name
17   location      = data.azurerm_resource_group.rg.location
18   resource_group_name = data.azurerm_resource_group.rg.name
19
20   security_rule {
21     name          = "Allow_SSH"
22     priority      = 1001
23     direction     = "Inbound"
24     access        = "Allow"
25     protocol      = "Tcp"
26     source_port_range = "*"
27     destination_port_range = "22"
28     source_address_prefix = "95.158.42.247/32"
29     destination_address_prefix = "*"
30   }
31
32   security_rule {
33     name          = "Allow_HTTP"
34     priority      = 1002
35     direction     = "Inbound"
36     access        = "Allow"
37     protocol      = "Tcp"
}

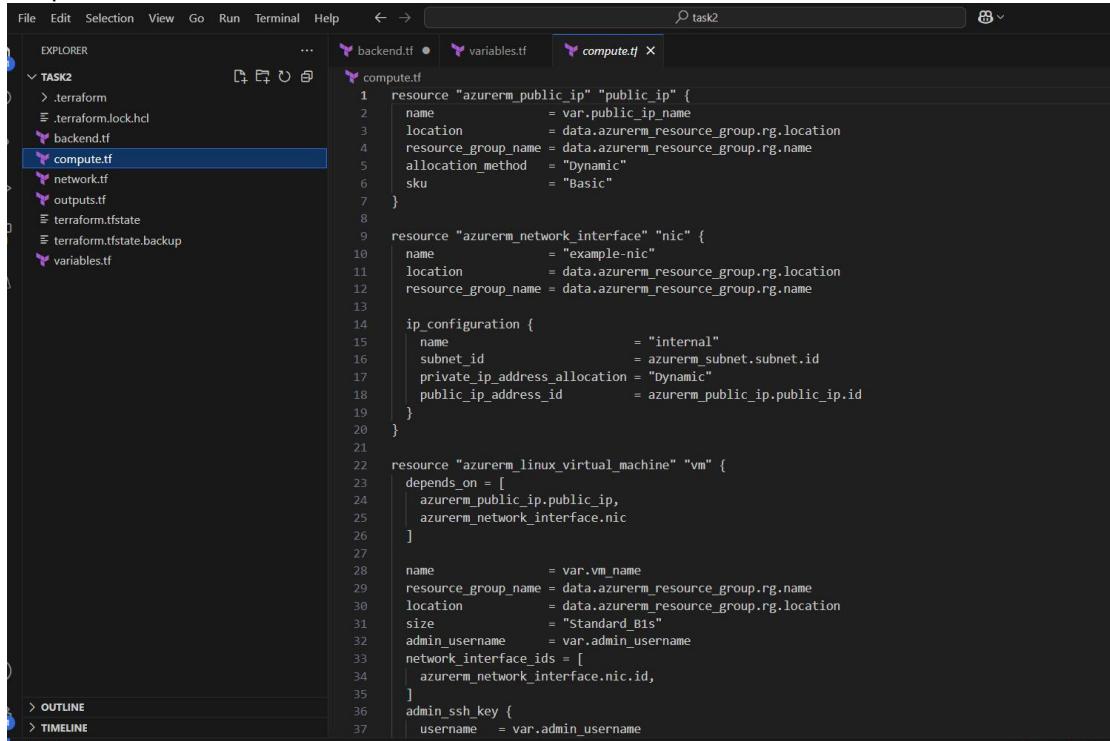
```

OUTLINE

TIMELINE

In 11 Col 41

compute.tf : VM

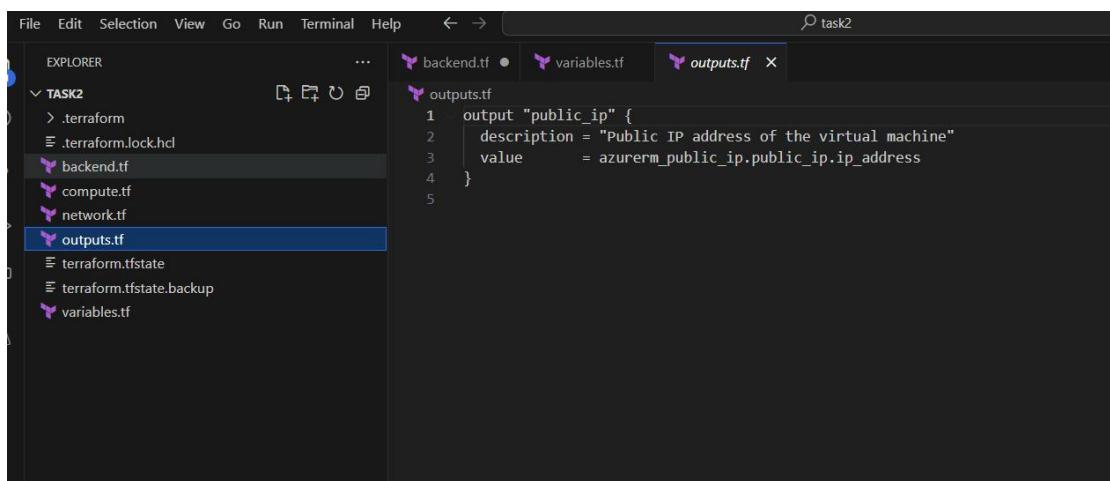


```
File Edit Selection View Go Run Terminal Help ⏪ ⏩ ⏴ ⏵ task2

EXPLORER ... backend.tf ● variables.tf compute.tf ✘
TASK2 > .terraform
> .terraform.lock.hcl
backend.tf
compute.tf
network.tf
outputs.tf
terraform.tfstate
terraform.tfstate.backup
variables.tf

1 resource "azurerm_public_ip" "public_ip" {
2   name          = var.public_ip_name
3   location      = data.azure_rm_resource_group.rg.location
4   resource_group_name = data.azure_rm_resource_group.rg.name
5   allocation_method = "Dynamic"
6   sku           = "Basic"
7 }
8
9 resource "azurerm_network_interface" "nic" {
10  name          = "example-nic"
11  location      = data.azure_rm_resource_group.rg.location
12  resource_group_name = data.azure_rm_resource_group.rg.name
13
14  ip_configuration {
15    name          = "internal"
16    subnet_id     = azurerm_subnet.subnet.id
17    private_ip_address_allocation = "Dynamic"
18    public_ip_address_id     = azurerm_public_ip.public_ip.id
19  }
20 }
21
22 resource "azurerm_linux_virtual_machine" "vm" {
23  depends_on = [
24    azurerm_public_ip.public_ip,
25    azurerm_network_interface.nic
26  ]
27
28  name          = var.vm_name
29  resource_group_name = data.azure_rm_resource_group.rg.name
30  location      = data.azure_rm_resource_group.rg.location
31  size          = "Standard_B1s"
32  admin_username = var.admin_username
33  network_interface_ids = [
34    azurerm_network_interface.nic.id,
35  ]
36  admin_ssh_key {
37    username = var.admin_username
38  }
39 }
40 }
```

outputs.tf : Public IP output



```
File Edit Selection View Go Run Terminal Help ⏪ ⏩ ⏴ ⏵ task2

EXPLORER ... backend.tf ● variables.tf outputs.tf ✘
TASK2 > .terraform
> .terraform.lock.hcl
backend.tf
compute.tf
network.tf
outputs.tf
terraform.tfstate
terraform.tfstate.backup
variables.tf

1 output "public_ip" {
2   description = "Public IP address of the virtual machine"
3   value        = azurerm_public_ip.public_ip.ip_address
4 }
5
```

Apply tf

```

PS E:\projects\terraformProj> task2> terraform apply
data.azurerm_resource_group.ng: Reading...
data.azurerm_resource_group.ng: Read complete after 0s [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova]
azurerm_public_ip.public_ip: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/publicIPAddresses/example-public-ip]
azurerm_virtual_network.vnet: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/example-vnet]
azurerm_network_security_group.ng: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/networkSecurityGroups/example-nsq]
azurerm_subnet.subnet: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/example-vnet/subnets/example-subnet]
azurerm_network_security_group_association.subnet_ns: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/example-vnet/subnets/example-subnet]
azurerm_network_interface.nic: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/networkInterfaces/example-nic]
azurerm_linux_virtual_machine.vm: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Compute/virtualMachines/example-vm]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

  # azurerm_public_ip.public_ip has changed
  ~ resource "azurerm_public_ip" "public_ip" {
      id          = "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/publicIPAddresses/example-public-ip"
      ip_address  = "9.163.1.90" -> "9.163.2.145"
      name        = "example-public-ip"
      tags         = {}
    }
    # (11 unchanged attributes hidden)

Unless you have made equivalent changes to your configuration, or ignored the relevant attributes using ignore_changes, the following plan may include actions to undo or respond to these changes.

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

```

```

Feb 10 23:22:16 example-vm systemd[1]: Starting A high performance web server and a reverse proxy server...
Feb 10 23:22:17 example-vm systemd[1]: nginx.service: Failed to parse PID from file /run/nginx.pid: Invalid argument
Feb 10 23:22:17 example-vm systemd[1]: Started A high performance web server and a reverse proxy server.
azureuser@example-vm:~$ curl http://40.68.12.213
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
azureuser@example-vm:~$ |

```

Practical Task 3: Implement a Scalable Infrastructure with Load Balancer and Auto Scaling

Requirements:

Extend the Terraform configuration to create a **highly available infrastructure** by deploying:

- A **Virtual Network (VNet)** with multiple subnets across **two Azure Availability Zones**.
 - An **Azure Load Balancer** with:
 - A **backend pool** of multiple **Virtual Machines (VMs)**.
 - A **health probe** for HTTP on port 80.
 - A **load-balancing rule** to distribute traffic across VMs.
 - A **Virtual Machine Scale Set (VMSS)** with:
 - At least **two VM instances** that auto-scale based on CPU usage.
 - A startup script to install **Apache** and deploy a sample website.
 - A **Storage Account** to store Terraform state remotely.
 - Verify that:
 - The **Load Balancer IP** distributes traffic between VM instances.
 - Auto-scaling works when CPU usage spikes.
 - Implement **Terraform modules** to modularize networking, compute, and security configurations.
 - Destroy the infrastructure when testing is complete.

1. Main configuration - main.tf

The screenshot shows a code editor interface with several tabs open, displaying Terraform configuration files. The left sidebar shows a tree view of the project structure:

- .terraform
- modules
- compute
- variables.tf
- loadbalancer
- main.tf
- variables.tf
- network
- main.tf
- outputs.tf
- variables.tf
- security
- main.tf
- variables.tf

The main editor area has tabs for `main.tf`, `variables.tf`, `outputs.tf`, `main.tf`, `main.tf`, `variables.tf`, and `variables.tf`. The `main.tf` tab is currently active and displays the following Terraform code:

```
provider "azurerm" {
  features {}
  subscription_id = "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
}

module "network" {
  source      = "./modules/network"
  location   = "westeurope"
  resource_group_name = "Volodymyr-Dibrova"
}

module "loadbalancer" {
  source      = "./modules/loadbalancer"
  resource_group_name = module.network.resource_group_name # Використовуємо output з модуля network
  location   = module.network.location # Використовуємо output з модуля network
}

module "compute" {
  source      = "./modules/compute"
  resource_group_name = module.network.resource_group_name
  location   = module.network.location
  subnet_id = module.network.subnet_id
  lb_backend_pool_id = module.loadbalancer.lb_backend_pool_id
  vmss_name = "example-vmss"
  lb_probe_id = module.loadbalancer.lb_probe_id # Передаємо lb_probe_id
  health_probe_id = module.loadbalancer.lb_probe_id # Передаємо health_probe_id
  lb_public_ip_id = module.loadbalancer.lb_public_ip_id # Нехаймо lb_public_ip_id
  min_instances = 1
  max_instances = 5
  instances = 2
}

output "lb_public_ip" {
  value = module.loadbalancer.lb_public_ip
}
output "vmss_id" {
  value = module.compute.vmss_id
}
```

modules/network/main.tf

The screenshot shows the Terraform IDE interface with the following components:

- EXPLORER**: Shows the project structure under **TASK3**, including `.terraform`, `modules`, `compute`, `loadbalancer`, `network`, and `security` modules, along with `main.tf`, `variables.tf`, `outputs.tf`, and `.terraform.lock.hcl`.
- MAIN.TF**: The main configuration file (`main.tf`) is open in the editor, showing Terraform code for creating a resource group, a virtual network, a subnet, a public IP, and a load balancer.
- OUTLINE**: A pane showing the outline of the current file.
- TIMELINE**: A pane showing the timeline of changes made to the file.

```
1 resource "azurerm_resource_group" "rg" {
2   name      = "Volodymyr-Dibrova"
3   location  = var.location
4 }
5
6 resource "azurerm_virtual_network" "vnet" {
7   name      = "example-vnet"
8   location  = var.location
9   resource_group_name = azurerm_resource_group.rg.name
10  address_space     = ["10.0.0.0/16"]
11 }
12
13 resource "azurerm_subnet" "subnet" {
14   name      = "subnet-vmss"
15   resource_group_name = azurerm_resource_group.rg.name
16   virtual_network_name = azurerm_virtual_network.vnet.name
17   address_prefixes  = ["10.0.1.0/24"]
18 }
19
20 resource "azurerm_public_ip" "lb_public_ip" {
21   name      = "example-lb-public-ip"
22   location  = var.location
23   resource_group_name = azurerm_resource_group.rg.name
24   allocation_method = "Static"
25   sku       = "Standard"
26 }
27
28 resource "azurerm_lb" "lb" {
29   name      = "example-lb"
30   location  = var.location
31   resource_group_name = azurerm_resource_group.rg.name
32   sku       = "Standard"
33
34   frontend_ip_configuration {
35     name      = "LoadBalancerFrontEnd"
36     public_ip_address_id = azurerm_public_ip.lb_public_ip.id
37 }
```

modules/loadbalancer/main.tf

The screenshot shows the VS Code interface with the Terraform extension. The Explorer sidebar on the left lists files and folders under the 'TASK3' task. The main editor area displays the 'main.tf' file for a load balancer, which defines resources like a public IP, a load balancer, and its backend pool, along with a probe and rules.

```
resource "azurerm_public_ip" "lb_public_ip" {
  name                = "example-lb-public-ip"
  location            = var.location
  resource_group_name = var.resource_group_name
  allocation_method   = "Static"
  sku                 = "Standard"
}

resource "azurerm_lb" "lb" {
  name                = "example-lb"
  location            = var.location
  resource_group_name = var.resource_group_name
  sku                 = "Standard"

  frontend_ip_configuration {
    name          = "LoadBalancerFrontEnd"
    public_ip_address_id = azurerm_public_ip.lb_public_ip.id
  }
}

resource "azurerm_lb_backend_address_pool" "lb_backend" {
  loadbalancer_id = azurerm_lb.lb.id
  name           = "lb-backend-pool"
}

resource "azurerm_lb_probe" "lb_probe" {
  loadbalancer_id = azurerm_lb.lb.id
  name           = "http-probe"
  port           = 80
  protocol       = "Http"
  request_path   = "/"
}

resource "azurerm_lb_rule" "lb_rule" {
  loadbalancer_id      = azurerm_lb.lb.id
  name                 = "http-rule"
  protocol             = "Tcp"
}
```

modules/compute/main.tf

The screenshot shows the VS Code interface with the Terraform extension. The Explorer sidebar on the left lists files and folders under the 'TASK3' task. The main editor area displays the 'main.tf' file for a VMSS, which defines a virtual machine scale set with specific configurations for the VMs.

```
resource "azurerm_windows_virtual_machine_scale_set" "vmss" {
  name                = var.vmss_name
  resource_group_name = var.resource_group_name
  location            = var.location
  health_probe_id     = var.health_probe_id
  overprovision        = true
  sku                 = "Standard_B1ms"
  instances           = var.instances # Замінили на змінну var.instances
  computer_name_prefix = "vmss" # Префікс не більше 9 символів

  source_image_reference {
    publisher = "MicrosoftWindowsServer"
    offer     = "WindowsServer"
    sku       = "2019-Datacenter"
    version   = "latest"
  }

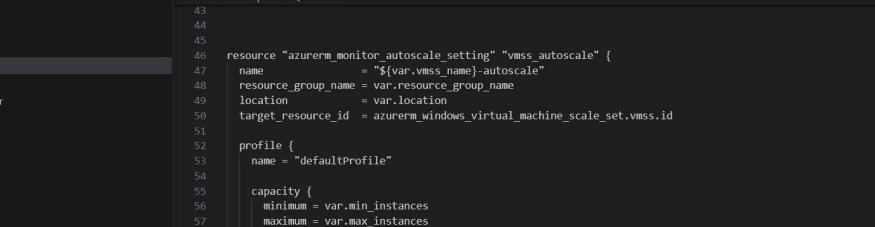
  admin_username = "azureuser"
  admin_password = "Me1122334455"

  os_disk {
    caching           = "ReadWrite"
    storage_account_type = "Standard_LRS"
  }

  network_interface {
    name      = "vmss-nic"
    primary   = true
  }

  ip_configuration {
    name           = "internal"
    subnet_id     = var.subnet_id
    load_balancer_backend_address_pool_ids = [var.lb_backend_pool_id]
    primary       = true
  }
}
```

Autoscale vmss



The screenshot shows a code editor with a dark theme. The left sidebar contains a tree view of the project structure:

- File
- Edit
- Selection
- View
- Go
- Run
- Terminal
- Help

Taskbar items include:

- task3
- File
- Edit
- Selection
- View
- Go
- Run
- Terminal
- Help

File Explorer pane:

- .terraform
- modules
- compute
- main.tf (selected)
- variables.tf
- loadbalancer
- main.tf
- variables.tf
- network
- main.tf
- outputs.tf
- variables.tf
- security
- main.tf
- variables.tf

Content pane (main.tf):

```
resource "azurerm_monitor_autoscale_setting" "vmss_autoscale" {
  name                = "${var.vmss_name}-autoscale"
  resource_group_name = var.resource_group_name
  location            = var.location
  target_resource_id  = azurerm_windows_virtual_machine_scale_set.vmss.id

  profile {
    name = "defaultProfile"

    capacity {
      minimum = var.min_instances
      maximum = var.max_instances
      default = var.instances
    }

    rule {
      metric_trigger {
        metric_name          = "Percentage CPU"
        metric_resource_id   = azurerm_windows_virtual_machine_scale_set.vmss.id
        time_grain           = "PT1M"
        statistic             = "Average"
        time_window           = "PT5M"
        time_aggregation     = "Average"
        operator              = "GreaterThan"
        threshold             = 75
      }

      scale_action {
        direction = "Increase"
        type      = "ChangeCount"
        value     = "+1"
        cooldown  = "PT5M"
      }
    }
  }
}
```

Apply tf

```
Windows PowerShell x + - 0

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

PS E:\projects\terraformProject\task3> terraform apply
module.network.azureurm_resource_group_rg: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova]
module.network.azureurm_public_ip_lb_public_ip: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/publicIPAddresses/example-lb-public-ip]
module.loadbalancer.azureurm_public_ip_lb_public_ip: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/publicIPAddresses/example-lb-public-ip]
module.network.azureurm_virtual_network_vnet: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/example-vnet]
module.network.azureurm_lb_lb: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb]
module.loadbalancer.azureurm_lb_lb: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb]
module.network.azureurm_subnet_subnet: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/example-vnet/subnets/subnet-vms]
module.loadbalancer.azureurm_lb_backend_address_pool_lb_backend: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/backendAddressPools/lb-backend-pool]
module.loadbalancer.azureurm_lb_probe_lb_probe: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/probes/http-probe]
module.network.azureurm_lb_probe_lb_probe: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/probes/http-probe]
module.loadbalancer.azureurm_lb_rule_lb_rule: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/loadBalancingRules/http-rule]
module.compute.azureurm_windows_virtual_machine_scale_set_vms: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Compute/virtualMachineScaleSets/example-vmss]
module.compute.azureurm_monitor_autoscale_setting_vmsas_autoscale: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Insights/autoScaleSettings/example-vmss-autoscale]
module.network.azureurm_lb_rule_lb_rule: Refreshing state... [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/loadBalancingRules/http-rule]

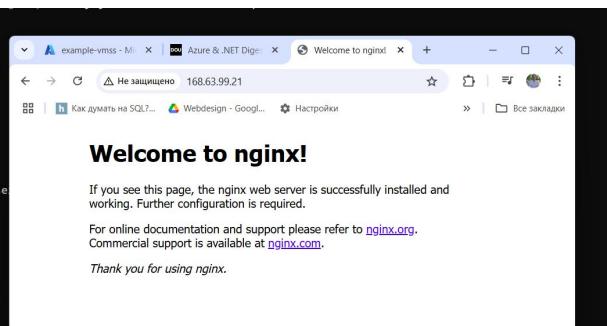
No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
```

```
PS E:\projects\terraformProj\task3> curl http://168.63.99.21
```

```
StatusCode : 200
StatusDescription : OK
Content : <!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif
}
</style>
<...
RawContent : HTTP/1.1 200 OK
Connection: keep-alive
Accept-Ranges: bytes
Content-Length: 612
Content-Type: text/html
Date: Tue, 11 Feb 2025 15:50:19 GMT
ETag: "67ab70b3-264"
Last-Modified: Tue, 11 Feb 2025 ...
Forms : []
Headers : [[Connection, keep-alive], [Accept-Ranges, bytes], [Content-Type, text/html]...]
Images : {}
InputFields : {}
Links : {{@innerHTML=nginx.org; innerText=nginx.org; outerHTML="<a href='http://nginx.org' tagName=A>"}, {@innerText=nginx.com; outerHTML="<a href='http://nginx.com' tagName=A>"; href="http://nginx.com/"}]}
ParsedHtml : System._ComObject
RawContentLength : 612
```



```

"zones": null
}
PS C:\WINDOWS\system32> az vmss list-instances --resource-group Volodymyr-Dibrova --name example-vmss --output table
InstanceID  LatestModelApplied  Location  ModelDefinitionApplied  Name  ProvisioningState  ResourceGroup  TimeCreated  VmId
----  -----  -----  -----  -----  -----  -----  -----  -----
2      True  westeurope  VirtualMachineScaleSet  example-vmss_2  Succeeded  Volodymyr-Dibrova  2025-02-11T23:35:22.504Z  597e7fe7-3612-4dca-87ca-95ac31172828
6      True  westeurope  VirtualMachineScaleSet  example-vmss_6  Succeeded  Volodymyr-Dibrova  2025-02-12T00:11:50.334Z  740e6ab9-3f4f-4ef3-a3c8-a34900b9c22c
PS C:\WINDOWS\system32>

```

health probe for HTTP on port 80

```

>> az network lb probe show --resource-group Volodymyr-Dibrova --lb-name example-lb --name http-probe
{
  "etag": "W/\"9dc1b5f-3514-4e58-8d29-6a10c545c72a\"",
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/probes/http-probe",
  "intervalInSeconds": 15,
  "loadBalancingRules": [
    {
      "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/loadBalancers/example-lb/loadBalancingRules/http-rule",
      "resourceGroup": "Volodymyr-Dibrova"
    }
  ],
  "name": "http-probe",
  "numberOfProbes": 2,
  "port": 80,
  "probeThreshold": 1,
  "protocol": "Http",
  "provisioningState": "Succeeded",
  "requestPath": "/",
  "resourceGroup": "Volodymyr-Dibrova",
  "type": "Microsoft.Network/loadBalancers/probes"
}
PS C:\WINDOWS\system32>
>> Get-Service W3SVC
Status   Name            DisplayName
-----  --  -----
Running  W3SVC           World Wide Web Publishing Service

```

```

PS E:\projects\terraformProj\task33> az storage blob list --container-name tfstate --account-name volodymyrtfstate01 --output table
There are no credentials provided in your command and environment, we will query for account key for your storage account.
It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials.
You also can add '--auth-mode login' in your command to use Azure Active Directory (Azure AD) for authorization if your login account is assigned required RBAC roles.
For more information about RBAC roles in storage, visit https://learn.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli.
In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get more information about environment variable usage.
Name          Blob Type  Blob Tier  Length  Content Type  Last Modified  Snapshot
tfstate       BlockBlob  Hot        35483   application/json  2025-02-11T23:40:26+00:00
PS E:\projects\terraformProj\task33> |

```

```

Windows PowerShell
PS E:\projects\terraformProj\task33> az storage blob download \
--container-name tfstate \
--name terraform.tfstate \
--file terraform.tfstate \
--account-name volodymyrtfstate01 \
--account-key f1rVF9ePjtr3aQhnJPoe8UykT3H9phsZvyKPz98RwdQ+lQ3R4YDjkpSV56kISAfngeAmFsMl3ot+AStuujqpQ==
Finished[########################################] 100. 0000%
{
  "container": "tfstate",
  "content": "",
  "contentMd5": null,
  "deleted": false,
  "encryptedMetadata": null,
  "encryptionKeySha256": null,
  "encryptionScope": null,
  "hasLegalHold": null,
  "hasVersionsOnly": null,
  "immutabilityPolicy": {
    "expiryTime": null,
    "policyMode": null
  },
  "isAppendBlobSealed": null,
  "isCurrentVersion": null,
  "lastAccessedOn": null,
  "metadata": {},
  "name": "terraform.tfstate",
  "objectReplicationDestinationPolicy": null,
  "objectReplicationSourceProperties": [],
  "properties": {
    "appendBlobCommittedBlockCount": null,
    "blobTier": null,
    "blobTierChangeTime": null,
    "blobTierInferred": null,
    "blobType": "BlockBlob",
    "contentLength": 35483,
    "contentRange": "bytes None-None/35483",
    "contentSettings": {
      "cacheControl": null,
      "contentDisposition": null,
      "contentEncoding": null,
      "contentLanguage": null
    }
  }
}

```

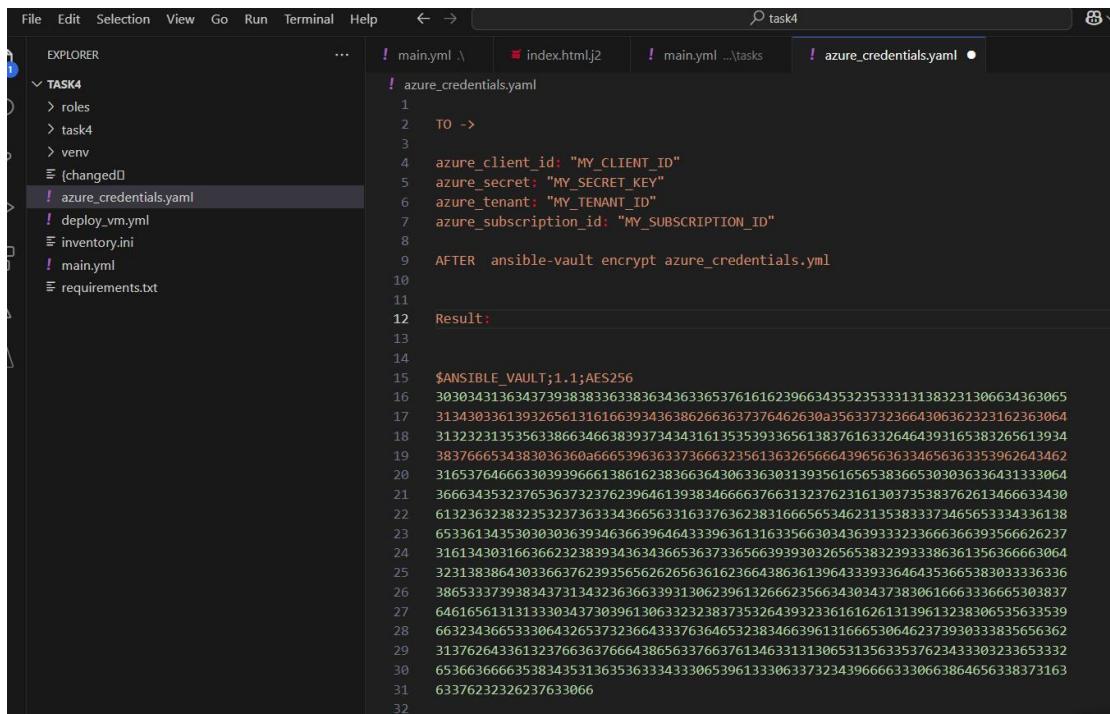
```

1 [{"version": 4,
2 "terraform_version": "1.10.4",
3 "serial": 1,
4 "lineage": "31c28877-d64e-2578-e658-1e385e99cd72",
5 "outputs": {},
6 "resources": [
7     {
8         "name": "example-vmss-autoscale",
9         "type": "azurerm_monitor_autoscale_setting",
10        "provider": "registry.terraform.io/hashicorp/azurerm\j",
11        "instances": [
12            {
13                "autoscale": "mobile.compute",
14                "managed": true,
15                "schema_version": 2,
16                "attributes": {
17                    "enabled": true,
18                    "id": "/subscriptions/9af8ee428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dirova/providers/Microsoft.Insights/autoScaleSettings/example-vmss-autoscale",
19                    "location": "westeurope",
20                    "name": "example-vmss-autoscale",
21                    "notification": [],
22                    "predictive": [],
23                    "profile": [
24                        {
25                            "capacity": [
26                                {
27                                    "default": 2,
28                                    "maximum": 5,
29                                    "minimum": 1
30                                }
31                            ],
32                            "fixed_date": [],
33                            "name": "defaultProfile",
34                            "recurrence": [],
35                            "rule": [
36                                {
37                                    "metric_trigger": [
38                                        {
39                                            "dimensions": []
40                                        }
41                                    ]
42                                }
43                            ]
44                        }
45                    ]
46                }
47            }
48        ]
49    }
50 }

```

Practical Task 4: Install and Configure Ansible for Azure Requirements:

- Install **Ansible** on your local machine.
- Verify the installation by checking the Ansible version.
- Install the **Azure Ansible Collection**.
- Authenticate Ansible with Azure using **service principal authentication**:
 - o Create an **Azure Active Directory Service Principal** with the necessary permissions.
 - o Retrieve the **client ID**, **tenant ID**, and **secret key**.
 - o Store credentials securely in an **Ansible Vault**.
- Write a basic **Ansible inventory file** that defines Azure as the target environment.
- Create a simple **Ansible playbook** that:
 - o Retrieves a list of all Azure resource groups.
 - o Prints the result as output.
 - o Execute the playbook and verify the output



```
task4
File Edit Selection View Go Run Terminal Help ← → ○ task4
EXPLORER ... ! main.yml \ index.htmlj2 ! main.yml ...\\tasks ! azure_credentials.yaml ●
! azure_credentials.yaml
1
2 TO ->
3
4 azure_client_id: "MY_CLIENT_ID"
5 azure_secret: "MY_SECRET_KEY"
6 azure_tenant: "MY_TENANT_ID"
7 azure_subscription_id: "MY_SUBSCRIPTION_ID"
8
9 AFTER ansible-vault encrypt azure_credentials.yaml
10
11
12 Result:
13
14
15 $ANSIBLE_VAULT;1.1;AES256
16 303034313634373938383633836343633653761616239663435323533131383231306634363065
17 3134303361393265613161663934363862663637376462630a356337323664306362323162363064
18 31323231353563386634663839373434316135353933656138376163326464393165383265613934
19 3837666534383036360a66665396363736663235613632656664396563634656363353962643462
20 316537646663303939666138616238663643063363031393561656538366530306336431333064
21 3666343523765363732376239646139383466663766313237623161303735383762613466633430
22 6132363238323532373633343665631633763623831666565346231353833377465653334336138
23 6533613435303030363934636639646433396316335663034363933233666366393566626237
24 3161340316636623238393436343653637365663939303265653832393386361356366663064
25 32313838643033663762393565626265636162366438636139643339336464353665383033336336
26 386533373938343731343236633931306239613266623566343034373830616663336665303837
27 646165613131330343730396130632323837353264393233616162613139613238306535633539
28 66323436653330643265373236643337636465328346639613166653064623739303383565362
29 313762643361323766363766643865633766376134631313065313563357623433303233653332
30 653663666635383435313635363334333065396133063373234396666333066386465633873163
31 63376232326237633066
32
```

retrieved a list of all Azure resource groups

EXPLORER

TASK4

- roles
- task4
- venv
- (changed) azure_credentials.yaml
- deploy_list_rg.yml
- inventory.ini
- main.yml
- requirements.txt

main.yml

```

1   ---
2   - name: Get a list of resource groups in Azure
3     hosts: localhost
4     gather_facts: no
5     vars_files:
6       - azure_credentials.yml
7     tasks:
8       - name: Get a list of resource groups
9         azure.azure_rm_resourcegroup_info:
10           client_id: "{{ azure_client_id }}"
11           secret: "{{ azure_secret }}"
12           tenant: "{{ azure_tenant }}"
13           subscription_id: "{{ azure_subscription_id }}"
14         register: rg_list
15
16       - name: View the full contents of the rg_list variable
17         debug:
18           var: rg_list
19
20       - name: List resource groups
21         debug:
22           msg: "{{ rg_list.resource_groups | default('No data on resource groups') }}"
23

```

vivi@DESKTOP-3QTOK67:~

```

vivi@DESKTOP-3QTOK67:~$ ansible --version
ansible [core: 2.12.0]
  config file = None
  configured module search path = ['/home/vivi/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/vivi/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Jan 17 2025, 18:03:48) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.13.2
  libyaml = True
vivi@DESKTOP-3QTOK67:~$ ansible-galaxy collection install azure.azcollection
Starting galaxy collection install process
Nothing to do, all requested collections are already installed. If you want to reinstall them, consider using '--force'.
vivi@DESKTOP-3QTOK67:~$ ansible-galaxy collection install azure.azcollection --force
Starting galaxy collection install process
Process Install dependency map
Starting collection install process
Downloading https://galaxy.ansible.com/api/v3/plugin/ansible/content/published/collections/artifacts/azure-azcollection-3.2.0.tar.gz to /home/vivi/.ansible/tmp/ansible-local-397206vxv0u4/tmpsqk6so98/azure-azcollection-3.2.0-gj08pxw
Installing 'azure.azcollection:3.2.0' to '/home/vivi/.ansible/collections/ansible_collections/azure/azcollection'
azure.azcollection:3.2.0 was installed successfully
vivi@DESKTOP-3QTOK67:~$ ansible-galaxy collection list azure.azcollection
# /home/vivi/.ansible/collections/ansible_collections
Collection          Version
-----
azure.azcollection  3.2.0

# /usr/lib/python3/dist-packages/ansible_collections
Collection          Version
-----
azure.azcollection  1.19.0

vivi@DESKTOP-3QTOK67:~$ pip install azure-cli azure-mgmt-resource
Command 'pip' not found, but can be installed with:
sudo apt install python3-pip
vivi@DESKTOP-3QTOK67:~$ az ad sp create-for-rbac --name "my-ansible-service-principal" --role Contributor --scopes /subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a8
Creating "Contributor" role assignment under scope '/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a8'
The output includes credentials that you must protect. Be sure that you do not include these credentials in your code or check the credentials into your source control. For more information, see https://aka.ms/adsp-clif
{
    "appId": "0f8add85-35b3-48e9-8e74-420d42105853",
    "displayName": "my-ansible-service-principal",
    "password": "VzZQ-2a1qBqWm1Kv-RtSDQkOsGKVlba03fpKbxh",
    "tenant": "8d1157b-1f96-415f-824b-ab0a29485d7d"
}
vivi@DESKTOP-3QTOK67:~$ 
```

Started Playbook

main.tf

index.html

azure_credentials.yaml

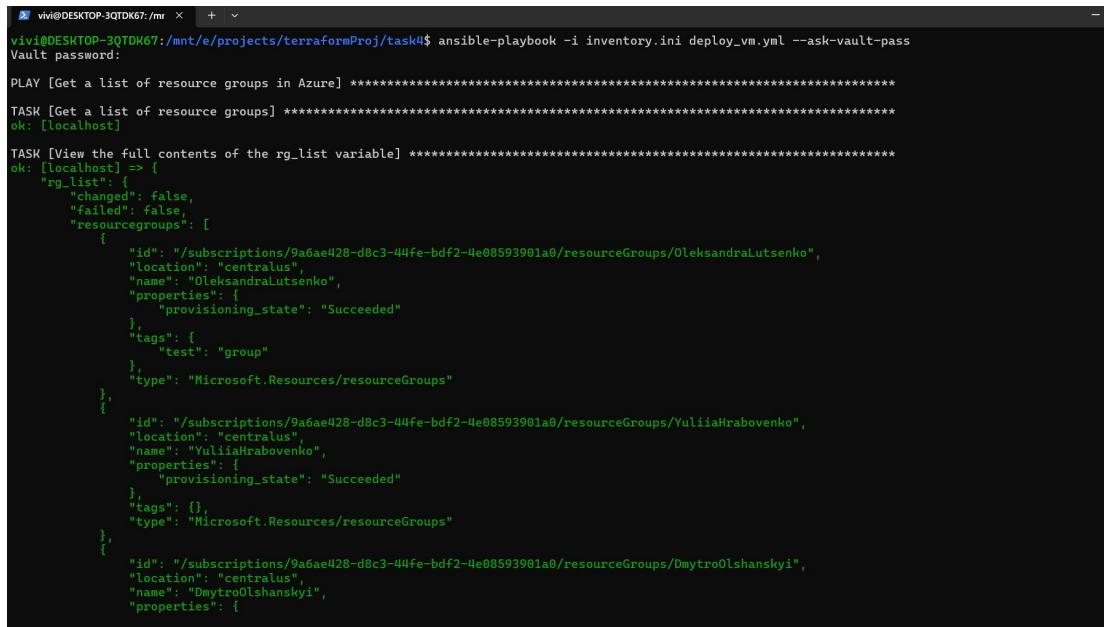
inventory.ini

deploy_vm.yml

```

1 [azure]
2   localhost ansible_connection=local ansible_python_interpreter=/mnt/e/projects/terraformProj/task4/venv/bin/python3
3 
```

Result:



```
vivi@DESKTOP-3QTDK67:/mnt/e/projects/terraformProj/task4$ ansible-playbook -i inventory.ini deploy_vm.yml --ask-vault-pass
Vault password:
PLAY [Get a list of resource groups in Azure] ****
TASK [Get a list of resource groups] ****
ok: [localhost]
ok: [localhost] => {
    "rg_list": [
        {
            "changed": false,
            "failed": false,
            "resourcegroups": [
                {
                    "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/OleksandraLutsenko",
                    "location": "centralus",
                    "name": "OleksandraLutsenko",
                    "properties": {
                        "provisioning_state": "Succeeded"
                    },
                    "tags": {
                        "test": "group"
                    },
                    "type": "Microsoft.Resources/resourceGroups"
                },
                {
                    "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/YuliiaHrabovenko",
                    "location": "centralus",
                    "name": "YuliiaHrabovenko",
                    "properties": {
                        "provisioning_state": "Succeeded"
                    },
                    "tags": {},
                    "type": "Microsoft.Resources/resourceGroups"
                },
                {
                    "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/DmytroOlshanskyi",
                    "location": "centralus",
                    "name": "DmytroOlshanskyi",
                    "properties": {

```

5.Practical Task 5: Deploy an Azure Virtual Machine with Ansible Requirements:

Extend the Ansible configuration to **deploy an Azure Virtual Machine (VM)**:

- Create an **Ansible playbook** that:
 - Defines an **Azure Virtual Network (VNet)** and a **Subnet**.
 - Creates a **Network Security Group (NSG)** with rules:
 - Allow **SSH (port 22)** inbound from a specified IP.
 - Allow **HTTP (port 80)** inbound for all users.
 - Deploys an **Ubuntu VM** in the subnet with an attached **public IP address**.
 - Uses **SSH key-based authentication** for the VM.
 - Write an **Ansible role** to configure the VM by:
 - Installing **Nginx** and starting the service.
 - Copying a custom **HTML file** to serve as the default web page.
 - Run the playbook and verify:
 - The VM is successfully deployed in Azure.
 - Nginx is running and the custom webpage is accessible via the public IP.
 - Implement **idempotency** by running the playbook multiple times and ensuring no unintended changes occur.

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under **TASKS**, including `ansible_project`, `group_vars` (with `vars.yml`), `roles\nginx_setup` (with `tasks` and `templates`), and `inventory.ini`. The file `main.yml` is currently selected.
- main.yml** (Ansible Project) tab: Displays the Ansible YAML code for deploying an Azure VM. The code includes tasks for creating a resource group, a virtual network, and a subnet.
- inventory.ini** (Ansible Project) tab: Shows the inventory configuration for the deployment.

```
- name: Deploy Azure VM with Ansible
hosts: localhost
connection: local
gather_facts: no

tasks:
- name: Ensure resource group exists
  azure.azcollection.azure_rm_resourcegroup:
    name: "Volodymyr-Dibrova"
    location: "West Europe"
- name: Create Virtual Network
  azure.azcollection.azure_rm_virtualnetwork:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleVNet"
    location: "East US"
    address_prefixes: "10.0.0.0/16"

- name: Create Subnet
  azure.azcollection.azure_rm_subnet:
    resource_group: "Volodymyr-Dibrova"
    virtual_network: "ansibleVNet"
    name: "ansibleSubnet"
    address_prefix: "10.0.1.0/24"
- name: Create Network Security Group
  azure.azcollection.azure_rm_securitygroup:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleNSG"
    location: "East US"
    rules:
      - name: AllowSSH
```

```
- name: Deploy Azure VM with Ansible
hosts: localhost
connection: local
gather_facts: no

tasks:
- name: Ensure resource group exists
  azure.azcollection.azure_rm_resourcegroup:
    name: "Volodymyr-Dibrova"
    location: "West Europe"
- name: Create Virtual Network
  azure.azcollection.azure_rm_virtualnetwork:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleVNet"
    location: "East US"
    address_prefixes: "10.0.0.0/16"

- name: Create Subnet
  azure.azcollection.azure_rm_subnet:
    resource_group: "Volodymyr-Dibrova"
    virtual_network: "ansibleVNet"
    name: "ansibleSubnet"
    address_prefix: "10.0.1.0/24"
- name: Create Network Security Group
  azure.azcollection.azure_rm_securitygroup:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleNSG"
    location: "East US"
    rules:
      - name: AllowSSH
```

```

        protocol: Tcp
        destination_port_range: 22
        access: Allow
        priority: 1000
        direction: Inbound
        source_address_prefix: "*"

    - name: AllowHTTP
      protocol: Tcp
      destination_port_range: 80
      access: Allow
      priority: 1001
      direction: Inbound
      source_address_prefix: "*"

- name: Create Public IP
  azure.azcollection.azure_rm_publicipaddress:
    resource_group: "Volodymyr-Dibrova"
    name: "ansiblePublicIP"
    location: "East US"
    allocation_method: Static

- name: Create Network Interface
  azure.azcollection.azure_rm_networkinterface:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleVM-nic"
    location: "East US"
    virtual_network: "ansibleVNet"
    subnet_name: "ansibleSubnet"
    security_group: "ansibleNSG"
    ip_configurations:
      - name: "ipconfig1"
        public_ip_name: "ansiblePublicIP"
        primary: true

- name: Create Ubuntu VM
  azure.azcollection.azure_rm_virtualmachine:
    resource_group: "Volodymyr-Dibrova"
    name: "ansibleVM"
    location: "East US"
    vm_size: "Standard_B1s"
    admin_username: "ansibleadmin"
    ssh_password_enabled: no
    ssh_public_keys:
      - path: "/home/ansibleadmin/.ssh/authorized_keys"
        key_data: "{{ lookup('file',
        '/mnt/c/Users/User/.ssh/id_rsa.pub') }}"
    image:
      offer: "UbuntuServer"
      publisher: "Canonical"
      sku: "18.04-LTS"
      version: "latest"
    network_interfaces:
      - name: "ansibleVM-nic"

- name: Configure Nginx on VM
  hosts: azure_vm
  become: yes
  tasks:

```

```

- name: Ensure python3-apt is installed
  raw: sudo apt-get update && sudo apt-get install -y python3-apt
  changed_when: false

- name: Install Nginx
  raw: sudo apt-get install -y nginx
  changed_when: false

- name: Copy custom index.html
  copy:
    content: "<h1>Welcome to my Ansible-managed Nginx server!</h1>"
    dest: /var/www/html/index.html
    owner: www-data
    group: www-data
    mode: '0644'

```

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows the project structure under "TASK4". The "main.yml" file in the "tasks" directory is currently selected.
- Code Editor:** Displays the "main.yml" file content for the "nginx_setup" role's tasks. It includes three tasks: installing Nginx, starting the service, and deploying a custom HTML page.
- Code Editor:** Displays the "index.html.j2" template file content, which contains the HTML code for the welcome page.
- Bottom Bar:** Shows the standard VS Code navigation bar with File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled "task4".

```

vivi@DESKTOP-3QTDK67:/mnt/e/projects/terraformProj/task4$ ansible-playbook -i inventory.ini main.yml
PLAY [Deploy Azure VM with Ansible] ****
TASK [Ensure resource group exists] ****
ok: [localhost]
TASK [Create Virtual Network] ****
ok: [localhost]
TASK [Create Subnet] ****
ok: [localhost]
TASK [Create Network Security Group] ****
ok: [localhost]
TASK [Create Public IP] ****
ok: [localhost]
TASK [Create Network Interface] ****
ok: [localhost]
TASK [Create Ubuntu VM] ****
ok: [localhost]
PLAY [Configure Nginx on VM] ****
TASK [Gathering Facts] ****
ok: [ansibleVM]
TASK [Ensure python3-apt is installed] ****
ok: [ansibleVM]
TASK [Install Nginx] ****
ok: [ansibleVM]
TASK [Copy custom index.html] ****
ok: [ansibleVM]

PLAY RECAP ****
ansibleVM : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost  : ok=7    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

(venv) vivi@DESKTOP-3QTDK67:/mnt/e/projects/terraformProj/task4$ 

```

Nginx працює

```

ansibleadmin@ansibleVM:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2025-02-12 22:53:13 UTC; 14min ago
     Docs: man:nginx(8)
 Main PID: 11982 (nginx)
    Tasks: 2 (limit: 1951)
   CGroup: /system.slice/nginx.service
           └─11982 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
              ├─11986 nginx: worker process

Feb 12 22:53:12 ansibleVM systemd[1]: Starting A high performance web server and a reverse proxy server...
Feb 12 22:53:13 ansibleVM systemd[1]: nginx.service: Failed to parse PID from file /run/nginx.pid: Invalid argument
Feb 12 22:53:13 ansibleVM systemd[1]: Started A high performance web server and a reverse proxy server.
ansibleadmin@ansibleVM:~$ 

```



Practical Task 7: Deploy a Resource Group Using an ARM Template Requirements:

- Create a **JSON-based ARM Template** that defines a **resource group** named **ARMResourceGroup** in the **East US** region.
- Use **az deployment sub create** to deploy the resource group using the ARM template.
- Verify the deployment in the Azure Portal or using the Azure CLI (**az group list**).
- Modify the template to add **tags** to the resource group and redeploy it.
- Remove the resource group after verification.

```

EXPLORER          ...
TASK7
arm-template.json 3
Welcome          arm-template.json 3 ...
0 arm-template.json > ...
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "resources": [
5     {
6       "type": "Microsoft.Resources/resourceGroups",
7       "apiVersion": "2021-04-01",
8       "location": "eastus",
9       "name": "ARMResourceGroup",
10      "properties": {}
11    }
12  ]
13 }

```

Розгортання ARM Template через Azure CLI

```

PS E:\projects\terraformProj\task7> az deployment sub create --location eastus --template-file arm-template.json
{
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/providers/Microsoft.Resources/deployments/arm-template",
  "location": "eastus",
  "name": "arm-template",
  "properties": {
    "correlationId": "32da2998-fe44-47eb-9850-a3feaae88d9f",
    "debugSetting": null,
    "dependencies": [],
    "duration": "PT3.0796238S",
    "error": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup"
      }
    ],
    "outputs": null,
    "parameters": null,
    "parametersLink": null,
    "providers": [
      {
        "id": null,
        "namespace": "Microsoft.Resources",
        "providerAuthorizationConsentState": null,
        "registrationPolicy": null,
        "registrationState": null,
        "resourceTypes": [
          {
            "aliases": null,
            "apiProfiles": null,
            "apiVersions": null,
            "capabilities": null,
            "defaultApiVersion": null,
            "locationMappings": null,
            "locations": [
              "eastus"
            ],
            ...
          }
        ]
      }
    ]
  }
}

```

Verifying the created resource group

```

PS E:\projects\terraformProj\task7> az group list --query "[?name=='ARMResourceGroup']"
[
  {
    "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup",
    "location": "eastus",
    "managedBy": null,
    "name": "ARMResourceGroup",
    "properties": {
      "provisioningState": "Succeeded"
    },
    "tags": null,
    "type": "Microsoft.Resources/resourceGroups"
  }
]
PS E:\projects\terraformProj\task7> |

```

Added tags

```

{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "resources": [
    {
      "type": "Microsoft.Resources/resourceGroups",
      "apiVersion": "2021-04-01",
      "location": "eastus",
      "name": "ARMResourceGroup",
      "properties": {},
      "tags": {
        "Project": "Terraform Task 7",
        "Environment": "Development",
        "Owner": "Volodymyr Dibrova"
      }
    }
  ]
}

```

Redeploying a resource group update

```

PS E:\projects\terraformProj\task7> az deployment sub create --location eastus --template-file arm-template.json
{
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/providers/Microsoft.Resources/deployments/arm-template",
  "location": "eastus",
  "name": "arm-template",
  "properties": {
    "correlationId": "c91ebdc4-3ac0-4795-9774-42e2858f4d1f",
    "debugSetting": null,
    "dependencies": [],
    "duration": "PT3.0287442S",
    "error": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup"
      }
    ],
    "outputs": null,
    "parameters": null,
    "parametersLink": null,
    "providers": [
      {
        "id": null,
        "namespace": "Microsoft.Resources",
        "providerAuthorizationConsentState": null,
        "registrationPolicy": null,
        "registrationState": null,
        "resourceTypes": [
          {
            "aliases": null,
            "apiProfiles": null,
            "apiVersions": null,
            "capabilities": null,
            "defaultApiVersion": null,
            "locationMappings": null,
            "locations": [
              "eastus"
            ],
            "properties": null,
            "resourceType": "resourceGroups",
            "zoneMappings": null
          }
        ]
      }
    ],
    "provisioningState": "Succeeded",
    "templateHash": "425857566789909578",
    "templateLink": null,
    "timestamp": "2025-02-12T23:42:22.404805+00:00",
    "validatedResources": null
  },
  "tags": null,
  "type": "Microsoft.Resources/deployments"
}
PS E:\projects\terraformProj\task7>

```

Check for tags

```

PS E:\projects\terraformProj\task7> az group show --name ARMResourceGroup --query "tags"
{
  "Environment": "Development",
  "Owner": "Volodymyr Dibrova",
  "Project": "Terraform Task 7"
}
PS E:\projects\terraformProj\task7>

```

Practical Task 8: Deploy an Azure Storage Account Using an ARM Template Requirements:

Extend the ARM template to define an **Azure Storage Account** with:

- A **unique name** and **StorageV2** account type.
- **Standard_LRS** as the replication type.

- Disable public access to the storage account by setting the public access level to Private for all containers.
- Add **parameters** for the storage account name and location.
- Deploy the template using **az deployment group create**.
- Validate the deployment using **az storage account list**.
- Modify the template to enable **Blob soft delete**, then redeploy it.
- Delete the storage account when done.

```

File Edit Selection View Go Run Terminal Help ⏎ ⏎ task8
EXPLORER TASK8 arm-rg-template.json > ...
    arm-rg-template.json
    arm-storage-template.json
1 [ {
2     "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3     "contentVersion": "1.0.0.0",
4     "parameters": {
5         "resourceGroupName": {
6             "type": "string",
7             "defaultValue": "ARMResourceGroup",
8             "metadata": {
9                 "description": "The name of the resource group."
10            }
11        },
12        "location": {
13            "type": "string",
14            "defaultValue": "eastus",
15            "metadata": {
16                "description": "Location for the resource group."
17            }
18        }
19    },
20    "resources": [
21        {
22            "type": "Microsoft.Resources/resourceGroups",
23            "apiVersion": "2021-04-01",
24            "name": "[parameters('resourceGroupName')]",
25            "location": "[parameters('location')]",
26            "properties": {},
27            "tags": {
28                "Environment": "Development",
29                "Owner": "Volodymyr Dibrova",
30                "Project": "Terraform Task 8"
31            }
32        }
33    ]
34 }
35

```

```

File Edit Selection View Go Run Terminal Help ⏎ ⏎ task8
EXPLORER TASK8 arm-rg-template.json 7 arm-storage-template.json 3
    arm-rg-template.json
    arm-storage-template.json
3 {
4     "contentVersion": "1.0.0.0",
5     "parameters": {
6         "storageAccountName": {
7             "type": "string",
8             "metadata": {
9                 "description": "The name of the storage account (must be globally unique)."
10            }
11        },
12        "location": {
13            "type": "string",
14            "defaultValue": "eastus",
15            "metadata": {
16                "description": "The location where the storage account will be deployed."
17            }
18        }
19    },
20    "resources": [
21        {
22            "type": "Microsoft.Storage/storageAccounts",
23            "apiVersion": "2022-09-01",
24            "name": "[parameters('storageAccountName')]",
25            "location": "[parameters('location')]",
26            "kind": "StorageV2",
27            "sku": {
28                "name": "Standard_LRS"
29            },
30            "properties": {
31                "accessTier": "Hot",
32                "allowBlobPublicAccess": false
33            }
34        }
35    ],
36    "outputs": {
37        "storageAccountId": {
38            "type": "string",
39            "value": "[resourceId('Microsoft.Storage/storageAccounts', parameters('storageAccountName'))]"
40        }
41    }
42 }
43

```

Created rg

```
PS E:\projects\terraformProj\task8> az deployment sub create `>>   --location eastus`>>   --template-file arm-rg-template.json`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/providers/Microsoft.Resources/deployments/arm-rg-template",`name": "arm-rg-template",`properties": {`correlationId": "e0228e11-793e-4fdf-a969-9690a42bc539",`debugSetting": null,`dependencies": [],`duration": "PT4.0086001S",`error": null,`mode": "Incremental",`onErrorDeployment": null,`outputResources": [`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup"`,`],`outputs": null,`parameters": {`location": {`type": "String",`value": "eastus"},`resourceGroupName": {`type": "String",`value": "ARMResourceGroup"},`},`parametersLink": null,`providers": [`{`id": null,`namespace": "Microsoft.Resources",`providerAuthorizationConsentState": null,`registrationPolicy": null,`registrationState": null,`resourceTypes": [`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Resources/deployments/arm-storage-template",`location": null,`name": "arm-storage-template",`properties": {`correlationId": "ae7cedd5-6056-47aa-a67b-111a77452398",`debugSetting": null,`dependencies": [],`duration": "PT22.0585648S",`error": null,`mode": "Incremental",`onErrorDeployment": null,`outputResources": [`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Storage/storageAccounts/mystorage641230478",`resourceGroup": "ARMResourceGroup"},`],`outputs": {`storageAccountId": {`type": "String",`value": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Storage/storageAccounts/mystorage641230478"},`},`parameters": {`location": {`type": "String",`value": "eastus"},`storageAccountName": {`type": "String",`value": "mystorage641230478"},`},`},`PS E:\projects\terraformProj\task8> az storage account check-name --name mystorage641230478`{`message": "The storage account named mystorage641230478 is already taken.",`nameAvailable": false,`reason": "AlreadyExists"},`}`
```

Created storage account

```
PS E:\projects\terraformProj\task8> az deployment group create `>>   --resource-group ARMResourceGroup`>>   --template-file arm-storage-template.json`>>   --parameters storageAccountName=mystorage$(Get-Random)`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Resources/deployments/arm-storage-template",`location": null,`name": "arm-storage-template",`properties": {`correlationId": "ae7cedd5-6056-47aa-a67b-111a77452398",`debugSetting": null,`dependencies": [],`duration": "PT22.0585648S",`error": null,`mode": "Incremental",`onErrorDeployment": null,`outputResources": [`{`id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Storage/storageAccounts/mystorage641230478",`resourceGroup": "ARMResourceGroup"},`],`outputs": {`storageAccountId": {`type": "String",`value": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/ARMResourceGroup/providers/Microsoft.Storage/storageAccounts/mystorage641230478"},`},`parameters": {`location": {`type": "String",`value": "eastus"},`storageAccountName": {`type": "String",`value": "mystorage641230478"},`},`},`PS E:\projects\terraformProj\task8> az storage account check-name --name mystorage641230478`{`message": "The storage account named mystorage641230478 is already taken.",`nameAvailable": false,`reason": "AlreadyExists"},`}`
```

Modified Soft Delete for Blob

```

{
  "parameters": {
    "storageAccountName": {
      "type": "string",
      "metadata": {
        "description": "The name of the storage account (must be globally unique)."
      }
    },
    "location": {
      "type": "string",
      "defaultValue": "eastus",
      "metadata": {
        "description": "The location where the storage account will be deployed."
      }
    }
  },
  "resources": [
    {
      "type": "Microsoft.Storage/storageAccounts",
      "apiVersion": "2022-09-01",
      "name": "[parameters('storageAccountName')]",
      "location": "[parameters('location')]",
      "kind": "StorageV2",
      "sku": {
        "name": "Standard_LRS"
      },
      "properties": {
        "accessTier": "Hot",
        "allowBlobPublicAccess": false,
        "deleteRetentionPolicy": {
          "enabled": true,
          "days": 7
        }
      }
    }
  ],
  "outputs": {
    "storageAccountId": {
      "type": "string",
      "value": "[resourceId('Microsoft.Storage/storageAccounts', parameters('storageAccountName'))]"
    }
  }
}

```

```

PS E:\projects\terraformProj\task8> az storage blob service-properties show --account-name mystorage641230478
There are no credentials provided in your command and environment, we will query for account key for your storage account.
It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials.
You also can add '--auth-mode login' in your command to use Azure Active Directory (Azure AD) for authorization if your login account has RBAC roles.
For more information about RBAC roles in storage, visit https://learn.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli.
In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get environment variable usage.
{
  "cors": [],
  "deleteRetentionPolicy": {
    "allowPermanentDelete": null,
    "days": 7,
    "enabled": true
  },
  "hourMetrics": {
    "enabled": true,
    "includeApis": true,
    "retentionPolicy": {
      "allowPermanentDelete": null,
      "days": 7,
      "enabled": true
    },
    "timeGrain": "1_0"
  }
}

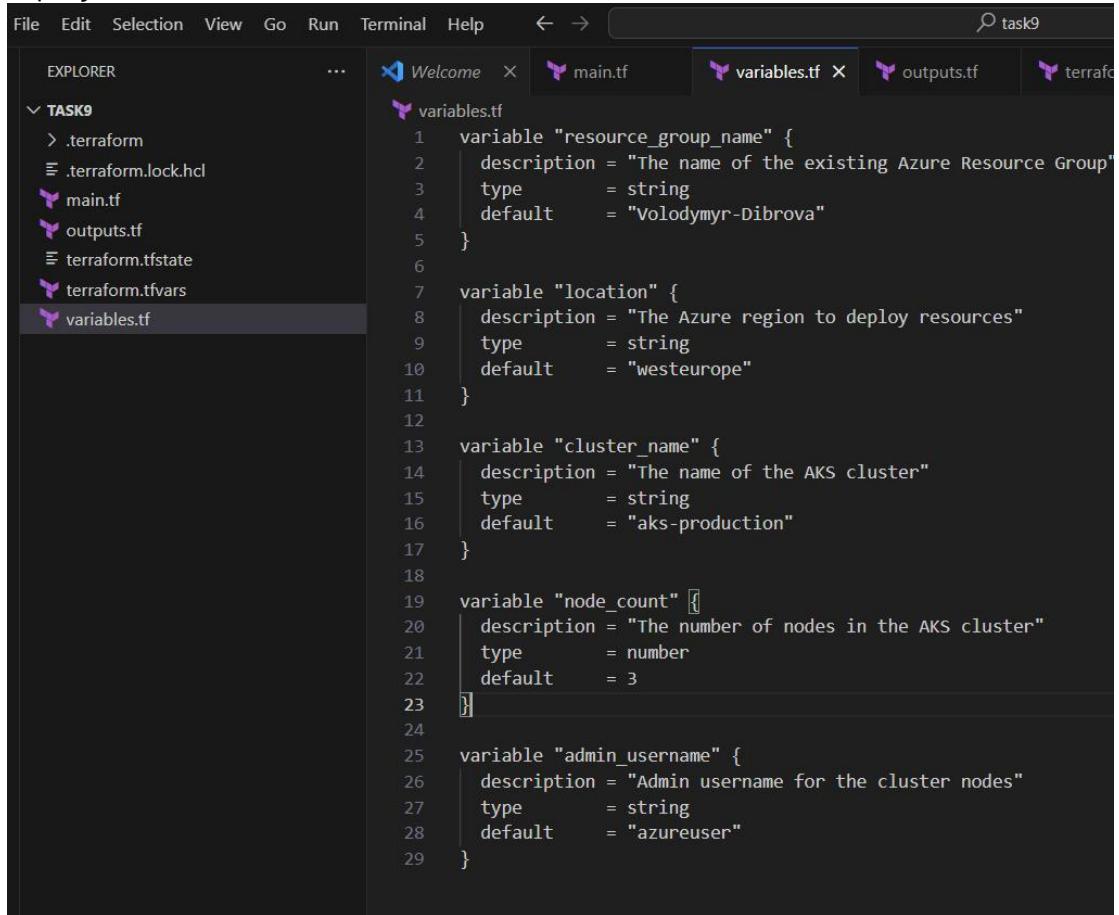
```

Practical Task 9: Terraform: Deploy a Production-Ready AKS Cluster with GitOps & Secret Management & Monitoring

1. Deploy an AKS cluster with Terraform, ensuring production readiness.
2. Integrate ArgoCD for GitOps-based application deployment.
3. Set up an Ingress Controller for ArgoCD access.
4. Enable Azure Key Vault CSI Driver for secure secrets management.
5. Deploy an application via ArgoCD, using a Git repository as the source.
6. Implement application health checks in ArgoCD using Kubernetes readiness/liveness probes.
7. Configure automated sync policies in ArgoCD to enable self-healing and pruning of outdated resources.
8. Enable Monitoring and Logging using:
 - Azure Monitor & Log Analytics for cluster-wide observability
 - Prometheus & Grafana for in-depth Kubernetes metrics

- Container Insights for real-time pod and node monitoring

Deployed an AKS cluster with Terraform



The screenshot shows a code editor interface with several tabs open. The tabs include 'Welcome', 'main.tf', 'variables.tf' (which is currently selected), 'outputs.tf', and 'terraform'. The 'EXPLORER' sidebar on the left shows files like '.terraform', '.terraform.lock.hcl', 'main.tf', 'outputs.tf', 'terraform.tfstate', 'terraform.tfvars', and 'variables.tf'. The 'variables.tf' tab contains the following Terraform code:

```

1  variable "resource_group_name" {
2    description = "The name of the existing Azure Resource Group"
3    type        = string
4    default     = "Volodymyr-Dibrova"
5  }
6
7  variable "location" {
8    description = "The Azure region to deploy resources"
9    type        = string
10   default    = "westeurope"
11 }
12
13 variable "cluster_name" {
14   description = "The name of the AKS cluster"
15   type        = string
16   default    = "aks-production"
17 }
18
19 variable "node_count" [
20   description = "The number of nodes in the AKS cluster"
21   type        = number
22   default    = 3
23 ]
24
25 variable "admin_username" {
26   description = "Admin username for the cluster nodes"
27   type        = string
28   default    = "azureuser"
29 }

```

```

File Edit Selection View Go Run Terminal Help ⏪ ⏴ task9
EXPLORER ... Welcome main.tf x variables.tf outputs.tf terraform.tfvars
TASK9 > .terraform < .terraform.lock.hcl
main.tf outputs.tf
> terraform.tfstate
> terraform.tfvars
variables.tf

main.tf
1 provider "azurerm" {
2   features {}
3   subscription_id = "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
4 }
5
6 data "azurerm_resource_group" "aks_rg" {
7   name = var.resource_group_name
8 }
9
10 resource "azurerm_kubernetes_cluster" "aks" {
11   name          = var.cluster_name
12   location      = data.azure_rm_resource_group.aks_rg.location
13   resource_group_name = data.azure_rm_resource_group.aks_rg.name
14   dns_prefix    = "aks-prod"
15
16   default_node_pool {
17     name        = "agentpool"
18     node_count = var.node_count
19     vm_size    = "Standard_D2_v2"
20   }
21
22   identity {
23     type = "SystemAssigned"
24   }
25
26   tags = {
27     environment = "production"
28     project    = "aks-gitops"
29   }
30 }
31
32 output "kube_config" {
33   value    = azurerm_kubernetes_cluster.aks.kube_config_raw
34   sensitive = true
35 }
36

```

> OUTLINE
i > TIMELINE

After terraform apply and checking state AKS

```

azurerm_kubernetes_cluster.aks: Still creating... [3m10s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [3m20s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [3m30s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [3m40s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [3m50s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [4m0s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [4m10s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [4m20s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [4m30s elapsed]
azurerm_kubernetes_cluster.aks: Still creating... [4m40s elapsed]
azurerm_kubernetes_cluster.aks: Creation complete after 4m47s [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.ContainerService/managedClusters/aks-production]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

kube_config = <sensitive>
PS E:\projects\terraformProj\task9> az aks show --resource-group Volodymyr-Dibrova --name aks-production --query "id"
"/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.ContainerService/managedClusters/aks-production"
PS E:\projects\terraformProj\task9> az aks get-credentials --resource-group Volodymyr-Dibrova --name aks-production
Merged "aks-production" as current context in C:\Users\User\.kube\config
PS E:\projects\terraformProj\task9> kubectl get nodes
NAME           STATUS  ROLES   AGE    VERSION
aks-agentpool-38728660-vmss000000  Ready   <none>  6m17s  v1.30.7
aks-agentpool-38728660-vmss000001  Ready   <none>  6m19s  v1.30.7
aks-agentpool-38728660-vmss000002  Ready   <none>  6m27s  v1.30.7
PS E:\projects\terraformProj\task9> |

```

Installed ArgoCD in AKS

```

PS E:\projects\terraformProj\task9> helm repo add argo https://argoproj.github.io/argo-helm
"argo" has been added to your repositories
PS E:\projects\terraformProj\task9> helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "argo" chart repository
Update Complete. ⚡Happy Helming!⚡
PS E:\projects\terraformProj\task9> kubectl create namespace argocd
namespace/argocd created
PS E:\projects\terraformProj\task9> helm install argocd argo/argo-cd --namespace argocd
NAME: argocd
LAST DEPLOYED: Thu Feb 13 14:46:12 2025
NAMESPACE: argocd
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
In order to access the server UI you have the following options:
1. kubectl port-forward service/argocd-server -n argocd 8080:443
   and then open the browser on http://localhost:8080 and accept the certificate
2. enable ingress in the values file 'server.ingress.enabled' and either
   - Add the annotation for ssl passthrough: https://argo-cd.readthedocs.io/en/stable/operator-manual/ingress/#option-1-ssl-passthrough
   - Set the 'configs.params."server.insecure"' in the values file and terminate SSL at your ingress: https://argo-cd.readthedocs.io/en/stable/operator-manual/ingress/#option-2-multiple-ingress-objects-and-hosts

After reaching the UI the first time you can login with username: admin and the random password generated during the installation. You can find the password by running:
kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d

(You should delete the initial secret afterwards as suggested by the Getting Started Guide: https://argo-cd.readthedocs.io/en/stable/getting-started/#4-logging-the-cluster)
PS E:\projects\terraformProj\task9> kubectl get nodes
NAME          STATUS   ROLES    AGE     VERSION
aks-agentpool-38728660-vmss000000  Ready   <none>   14m    v1.38.7
aks-agentpool-38728660-vmss000001  Ready   <none>   14m    v1.38.7
aks-agentpool-38728660-vmss000002  Ready   <none>   14m    v1.38.7
PS E:\projects\terraformProj\task9>

```

Configuring access via Ingress Controller

```

PS E:\projects\terraformProj\task9> kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/main/deploy/static/provider/cloud/deployment.yaml
namespace/ingress-nginx created
serviceaccount/ingress-nginx created
serviceaccount/ingress-nginx-admission created
role.rbac.authorization.k8s.io/ingress-nginx created
role.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrole.rbac.authorization.k8s.io/ingress-nginx created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission created
rolebinding.rbac.authorization.k8s.io/ingress-nginx created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
configmap/ingress-nginx-controller created
service/ingress-nginx-controller created
service/ingress-nginx-controller-admission created
deployment.apps/ingress-nginx-controller created
job.batch/ingress-nginx-admission-create created
job.batch/ingress-nginx-admission-patch created
ingressclass.networking.k8s.io/nginx created
validatingwebhookconfiguration.admissionregistration.k8s.io/ingress-nginx-admission created

due to a change in configuration
PS E:\projects\terraformProj\task9> kubectl get all -n ingress-nginx
NAME                           READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-admission-create-pl4x6   0/1    Completed   0          118s
pod/ingress-nginx-admission-patch-59nlb    0/1    Completed   0          118s
pod/ingress-nginx-controller-cbb88bdbc-lmr4v4 1/1    Running    0          118s

NAME                         TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)           AGE
service/ingress-nginx-controller   LoadBalancer   10.0.226.110  132.164.41.220  80:31464/TCP,443:32125/TCP   119s
service/ingress-nginx-controller-admission ClusterIP   10.0.251.114  <none>        443/TCP          119s

NAME                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ingress-nginx-controller   1/1    1           1           2m

NAME                           DESIRED  CURRENT  READY   AGE
replicaset.apps/ingress-nginx-controller-cbb88bdbc  1       1       1       119s

NAME                           STATUS    COMPLETIONS  DURATION   AGE
job.batch/ingress-nginx-admission-create   Complete  1/1        7s         119s
job.batch/ingress-nginx-admission-patch   Complete  1/1        7s         119s
PS E:\projects\terraformProj\task9> kubectl get pods -n argocd
NAME                           READY   STATUS    RESTARTS   AGE
argocd-application-controller-0   1/1    Running   0          7m47s
argocd-applicationset-controller-5ff8f4455c-cr6rs  1/1    Running   0          7m47s
argocd-dex-server-7f5bb4d7f8-n854r  1/1    Running   0          7m47s
argocd-notifications-controller-bc974676c-dsfh8  1/1    Running   0          7m47s
argocd-redis-79965fc87d-4f44q  1/1    Running   0          7m47s
argocd-repo-server-576bc4b5d5-mnkz9  1/1    Running   0          7m47s
argocd-server-7f76c945bb-hptkb  1/1    Running   0          7m47s
PS E:\projects\terraformProj\task9> kubectl get pods -n ingress-nginx
NAME                           READY   STATUS    RESTARTS   AGE
ingress-nginx-admission-create-pl4x6   0/1    Completed   0          3m51s
ingress-nginx-admission-patch-59nlb    0/1    Completed   0          3m51s
ingress-nginx-controller-cbb88bdbc-lmr4v4 1/1    Running    0          3m51s
PS E:\projects\terraformProj\task9> kubectl create namespace argocd
Error from server (AlreadyExists): namespaces "argocd" already exists

```

Configured Ingress for ArgoCD

```

EXPLORER          ...
TASK9             ...
> .terraform
≡ .terraform.lock.hcl
! ingress-argocd.yaml
main.tf
outputs.tf
≡ terraform.tfstate
terraform.tfvars
variables.tf

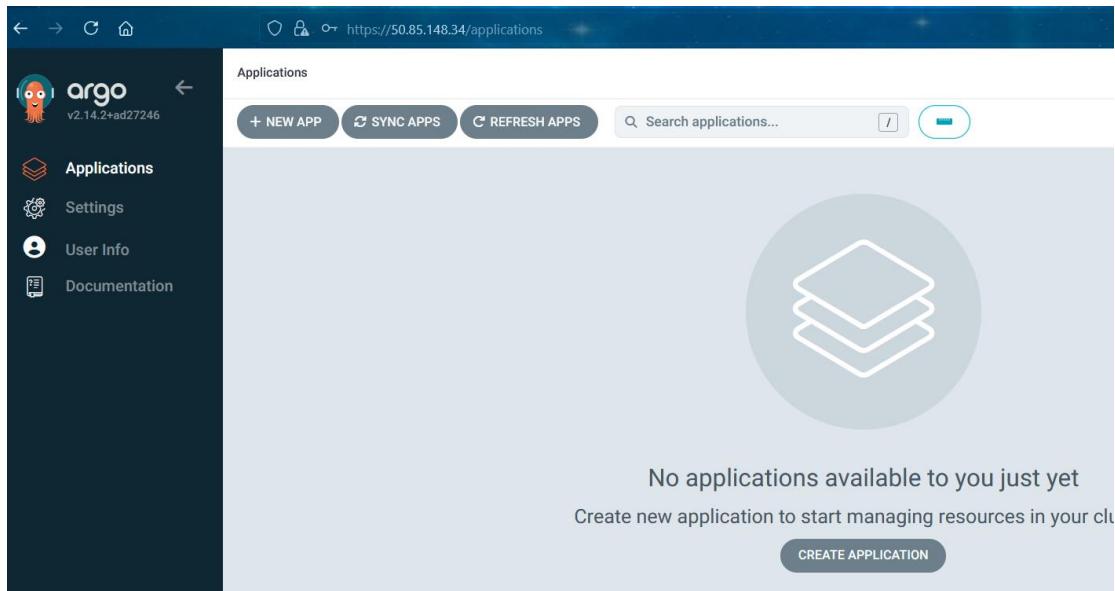
Welcome          ...
main.tf           ...
! ingress-argocd.yaml X
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: argocd-server
5    namespace: argocd
6  spec:
7    type: LoadBalancer
8    ports:
9      - port: 443
10     targetPort: 8080
11     protocol: TCP
12     name: https
13   selector:
14     app.kubernetes.io/name: argocd-server
15

```

```

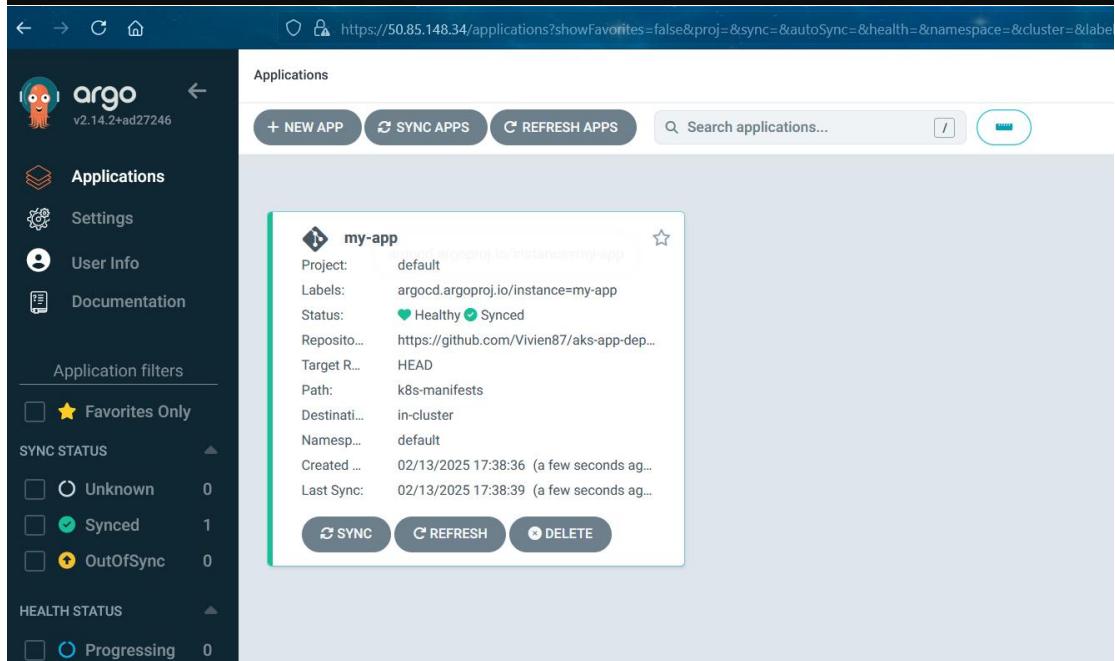
PS E:\projects\terraformProj\task9> kubectl apply -f ingress-argocd.yaml
service/argocd-server created
PS E:\projects\terraformProj\task9> kubectl get svc -n argocd
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP  PORT(S)         AGE
argocd-applicationset-controller   ClusterIP  10.0.227.30 <none>       7000/TCP        44m
argocd-dex-server                 ClusterIP  10.0.253.5  <none>       5556/TCP,5557/TCP  44m
argocd-redis                      ClusterIP  10.0.251.205 <none>       6379/TCP        44m
argocd-repo-server                ClusterIP  10.0.55.110 <none>       8081/TCP        44m
argocd-server                     LoadBalancer 10.0.20.66  <pending>   443:31322/TCP   7s
PS E:\projects\terraformProj\task9> kubectl get svc -n argocd
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP  PORT(S)         AGE
argocd-applicationset-controller   ClusterIP  10.0.227.30 <none>       7000/TCP        44m
argocd-dex-server                 ClusterIP  10.0.253.5  <none>       5556/TCP,5557/TCP  44m
argocd-redis                      ClusterIP  10.0.251.205 <none>       6379/TCP        44m
argocd-repo-server                ClusterIP  10.0.55.110 <none>       8081/TCP        44m
argocd-server                     LoadBalancer 10.0.20.66  <pending>   443:31322/TCP   12s
PS E:\projects\terraformProj\task9> kubectl get svc -n argocd
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP  PORT(S)         AGE
argocd-applicationset-controller   ClusterIP  10.0.227.30 <none>       7000/TCP        45m
argocd-dex-server                 ClusterIP  10.0.253.5  <none>       5556/TCP,5557/TCP  45m
argocd-redis                      ClusterIP  10.0.251.205 <none>       6379/TCP        45m
argocd-repo-server                ClusterIP  10.0.55.110 <none>       8081/TCP        45m
argocd-server                     LoadBalancer 10.0.20.66  50.85.148.34  443:31322/TCP   45s
PS E:\projects\terraformProj\task9> |
PS E:\projects\terraformProj\task9> kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath=".data.password" | Out-String | %{$([System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($_.Trim())))}
$UrU68w0IBAvrx7
PS E:\projects\terraformProj\task9> kubectl get pods -n argocd
NAME                           READY   STATUS    RESTARTS   AGE
argocd-application-controller-0  1/1    Running   0          20m
argocd-applicationset-controller-5ff8f4455c-cr6rs  1/1    Running   0          20m
argocd-dex-server-7f5bb4d7f8-n854r  1/1    Running   0          20m
argocd-notifications-controller-bc974676c-dsfh8  1/1    Running   0          20m
argocd-redis-79965fc87d-4ff4q  1/1    Running   0          20m
argocd-repo-server-576bc4b5d5-mnkz9  1/1    Running   0          20m
argocd-server-7f76c945bb-hptkb  1/1    Running   0          20m
PS E:\projects\terraformProj\task9> kubectl get svc -n argocd
NAME            TYPE        CLUSTER-IP   EXTERNAL-IP  PORT(S)         AGE
argocd-applicationset-controller   ClusterIP  10.0.227.30 <none>       7000/TCP        21m
argocd-dex-server                 ClusterIP  10.0.253.5  <none>       5556/TCP,5557/TCP  21m
argocd-redis                      ClusterIP  10.0.251.205 <none>       6379/TCP        21m
argocd-repo-server                ClusterIP  10.0.55.110 <none>       8081/TCP        21m
argocd-server                     ClusterIP  10.0.117.104 <none>       88/TCP,443/TCP   21m
PS E:\projects\terraformProj\task9> kubectl port-forward svc/argocd-server -n argocd 8080:443
Forwarding from 127.0.0.1:8080 -> 8080
Forwarding from [::]:8080 -> 8080
Handling connection for 8080
E0213 15:08:03.080065 10620 portforward.go:398] error copying from local connection to remote stream: writeto tcp4 127.0.0.1:8080->127.0.0.1:22508: read t
cp4 127.0.0.1:8080->127.0.0.1:22508: wsarecv: An existing connection was forcibly closed by the remote host.
Handling connection for 8080

```



Deploy an application via ArgoCD, using a Git repository as the source.

```
PS E:\projects\terraformProj\task9\1\aks-app-deploy> argo cd app create my-app ` 
--repo https://github.com/Vivien87/aks-app-deploy.git 
--path k8s-manifests 
--dest-server https://kubernetes.default.svc 
--dest-namespace default 
--sync-policy automated --upsert 
application my-app updated 
PS E:\projects\terraformProj\task9\1\aks-app-deploy> argo cd app list 
NAME CLUSTER NAMESPACE PROJECT STATUS HEALTH SYNCPOLICY CONDITIONS REPO 
argocd/my-app https://kubernetes.default.svc default default Synced Healthy Auto-Prune <none> https://github.com/Vivien87/aks-app-deploy.git 
k8s-manifests HEAD 
PS E:\projects\terraformProj\task9\1\aks-app-deploy> argo cd app get my-app 
Name: argocd/my-app 
Project: default 
Server: https://kubernetes.default.svc 
Namespace: default 
URL: https://argocd.example.com/applications/my-app 
Source: 
- Repo: https://github.com/Vivien87/aks-app-deploy.git 
Target: HEAD 
Path: k8s-manifests 
SyncWindow: Sync Allowed 
Sync Policy: Automated (Prune) 
Sync Status: Synced to HEAD (5e839cd) 
Health Status: Healthy 
GROUP KIND NAMESPACE NAME STATUS HEALTH HOOK MESSAGE 
Service argo cd argo cd-server Synced Healthy service/argo cd-server unchanged 
argo proj.io Application argo cd my-app Synced application.argo proj.io/my-app configured 
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl get ingress -n argo cd 
NAME CLASS HOSTS ADDRESS PORTS AGE 
argo cd-ingress nginx argo cd example.com 132.164.41.220 80 6h55m 
PS E:\projects\terraformProj\task9\1\aks-app-deploy>
```



Enable Azure Key Vault CSI Driver for secure secrets management.

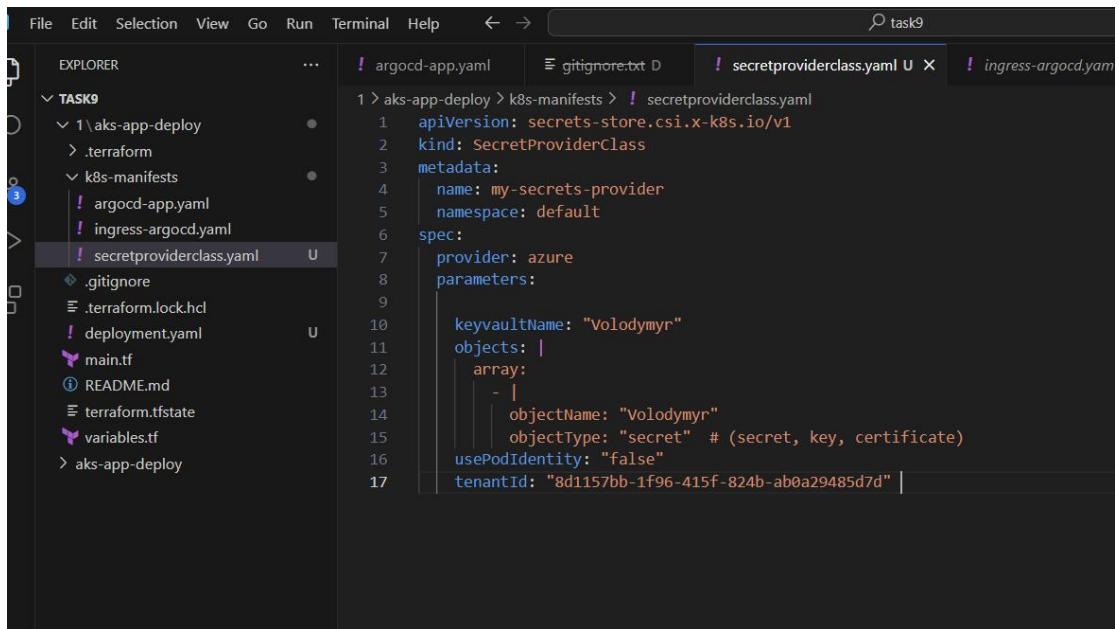
```
PS E:\projects\terraformProj\task9\1\aks-app-deploy> helm install csi-secrets-store secrets-store-csi-driver/secrets-store-csi-driver --namespace kube-system --replace
NAME: csi-secrets-store
LAST DEPLOYED: Thu Feb 13 19:13:23 2025
NAMESPACE: kube-system
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The Secrets Store CSI Driver is getting deployed to your cluster.

To verify that Secrets Store CSI Driver has started, run:

  kubectl --namespace=kube-system get pods -l "app=secrets-store-csi-driver"

Now you can follow these steps https://secrets-store-csi-driver.sigs.k8s.io/getting-started/usage.html
to create a SecretProviderClass resource, and a deployment using the SecretProviderClass.

PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl --namespace=kube-system get pods -l "app=secrets-store-csi-driver"
NAME                                         READY   STATUS    RESTARTS   AGE
csi-secrets-store-secrets-store-csi-driver-dzf7s  3/3    Running   0          2m40s
csi-secrets-store-secrets-store-csi-driver-f86mm  3/3    Running   0          2m40s
csi-secrets-store-secrets-store-csi-driver-sz4c2  3/3    Running   0          2m40s
```



The screenshot shows a code editor interface with the title bar "task9". The left sidebar displays a file tree for a project named "TASK9" containing subfolders "1\aks-app-deploy" and "k8s-manifests", along with files ".gitignore", ".terraform.lock.hcl", "deployment.yaml", "main.tf", "README.md", "terraform.tfstate", "variables.tf", and "aks-app-deploy". The right pane shows the content of the "secretproviderclass.yaml" file:

```
1 > aks-app-deploy > k8s-manifests > secretproviderclass.yaml
  1 apiVersion: secrets-store.csi.x-k8s.io/v1
  2 kind: SecretProviderClass
  3 metadata:
  4   name: my-secrets-provider
  5   namespace: default
  6 spec:
  7   provider: azure
  8   parameters:
  9     keyvaultName: "Volodymyr"
 10   objects: |
 11     array:
 12       -
 13         objectName: "Volodymyr"
 14         objectType: "secret" # (secret, key, certificate)
 15         usePodIdentity: "false"
 16         tenantId: "8d1157bb-1f96-415f-824b-ab0a29485d7d"
```

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-app
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      app: my-app
  template:
    metadata:
      labels:
        app: my-app
    spec:
      volumes:
        - name: secrets-store
          csi:
            driver: secrets-store.csi.k8s.io
            readOnly: true
            volumeAttributes:
              secretProviderClass: "my-secrets-provider"
      containers:
        - name: my-app-container
          image: busybox
          command: ["sleep", "3600"]
          volumeMounts:
            - name: secrets-store
              mountPath: /mnt/secrets-store

```

```

PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl logs my-app-695d5df64f-l5jgv
error: error from server (NotFound): pods "my-app-695d5df64f-l5jgv" not found in namespace "default"
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
csi-secrets-store-csi-driver-7mqq5  3/3     Running   0          3h30m
csi-secrets-store-secrets-store-csi-driver-9ngnl  3/3     Running   0          3h30m
csi-secrets-store-secrets-store-csi-driver-dqnzs  3/3     Running   0          3h30m
my-app-695d5df64f-9sv44   0/1     ContainerCreating   0          23s
my-app-6cb9f94d59-bfmfz   0/1     ContainerCreating   0          18m
PS E:\projects\terraformProj\task9\1\aks-app-deploy> helm install csi-secrets-store secrets-store-csi-driver/secrets-store-csi-driver --namespace kube-system
Error: INSTALLATION FAILED: cannot re-use a name that is still in use

```

Implement application health checks in ArgoCD using Kubernetes readiness/liveness probes.

```

File Edit Selection View Go Run Terminal Help ← → ⌂ task9

EXPLORER ... argocd-app.yaml gitignore.txt D secretproviderclass.yaml U in
TASK9 1\aks-app-deploy .terraform k8s-manifests
  argocd-app.yaml ingress-argocd.yaml secretproviderclass.yaml .gitignore .terraform.lock.hcl deployment.yaml U
  main.tf README.md terraform.tfstate variables.tf aks-app-deploy

1 > aks-app-deploy > deployment.yaml
6   spec:
8     selector:
11    template:
12      metadata:
13        labels:
14          app: my-app
15        spec:
16          containers:
17            - name: my-app-container
18              image: busybox
19              command: ["sleep", "3600"]
20              volumeMounts:
21                - name: secrets-store
22                  mountPath: /mnt/secrets-store
23              livenessProbe:
24                httpGet:
25                  path: /healthz
26                  port: 8080
27                  initialDelaySeconds: 10
28                  periodSeconds: 5
29              readinessProbe:
30                httpGet:
31                  path: /readiness
32                  port: 8080
33                  initialDelaySeconds: 5
34                  periodSeconds: 5
35              volumes:
36                - name: secrets-store
37                  csi:
38                    driver: secrets-store.csi.k8s.io

PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
csi-secrets-store-csi-driver-7mqq5  3/3    Running   0          3h52m
csi-secrets-store-csi-driver-9ngnl  3/3    Running   0          3h52m
csi-secrets-store-csi-driver-dqnzs  3/3    Running   0          3h52m
my-app-6cb9f94d59-bfmfz             0/1    ContainerCreating   0          40m
my-app-7d4759f9d9-xqfmj            0/1    ContainerCreating   0          8m54s

PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl describe pod my-app-7d4759f9d9-xqfmj
Name:         my-app-7d4759f9d9-xqfmj
Namespace:    default
Priority:    0
Service Account: default
Node:        aks-agentpool-38728660-vmss000002/10.224.0.6
Start Time:   Thu, 13 Feb 2025 21:59:48 +0200
Labels:      app=my-app
Annotations: pod-template-hash=7d4759f9d9
Status:      Pending
IP:          <none>
Controlled By: ReplicaSet/my-app-7d4759f9d9
Containers:
  my-app-container:
    Container ID:   docker://busybox
    Image:          busybox
    Image ID:       <none>
    Port:           <none>
    Host Port:     <none>
    Command:
      sleep
      3600
    State:          Waiting
      Reason:        ContainerCreating
    Ready:          False
    Restart Count:  0
    Liveness:       http-get http://:8080/healthz delay=10s timeout=1s period=5s #success=1 #failure=3
    Readiness:      http-get http://:8080/readiness delay=5s timeout=1s period=5s #success=1 #failure=3
    Environment:   <none>
    Mounts:
      /mnt/secrets-store from secrets-store (rw)

```

7. Configure automated sync policies in ArgoCD to enable self-healing and pruning of outdated resources.

```

PS E:\projects\terraformProj\task9\1\aks-app-deploy> argocd app set my-app \
>>   --sync-policy automated
>>   --auto-prune
>>   --self-heal
PS E:\projects\terraformProj\task9\1\aks-app-deploy> argocd app get my-app
Name:           argocd/my-app
Project:        default
Server:         https://kubernetes.default.svc
Namespace:      default
URL:           https://argocd.example.com/applications/my-app
Source:
- Repo:          https://github.com/Vivien87/aks-app-deploy.git
  Target:        HEAD
  Path:          k8s-manifests
SyncWindow:     Sync Allowed
Sync Policy:    Automated (Prune)
Sync Status:    Synced to HEAD (5e839cd)
Health Status: Healthy

GROUP      KIND      NAMESPACE  NAME      STATUS  HEALTH  HOOK  MESSAGE
Service    argocd    argocd-server Synced  Healthy   service/argocd-server unchanged
argoproj.io Application argocd    my-app   Synced   application.argoproj.io/my-app configured
PS E:\projects\terraformProj\task9\1\aks-app-deploy>

```

7. Configure automated sync policies in ArgoCD to enable self-healing and pruning of outdated resources.

```

PS E:\projects\terraformProj\task9\1\aks-app-deploy> argocd app set my-app \
>>   --sync-policy automated
>>   --auto-prune
>>   --self-heal
PS E:\projects\terraformProj\task9\1\aks-app-deploy> argocd app get my-app
Name:           argocd/my-app
Project:        default
Server:         https://kubernetes.default.svc
Namespace:      default
URL:           https://argocd.example.com/applications/my-app
Source:
- Repo:          https://github.com/Vivien87/aks-app-deploy.git
  Target:        HEAD
  Path:          k8s-manifests
SyncWindow:     Sync Allowed
Sync Policy:    Automated (Prune)
Sync Status:    Synced to HEAD (5e839cd)
Health Status: Healthy

GROUP      KIND      NAMESPACE  NAME      STATUS  HEALTH  HOOK  MESSAGE
Service    argocd    argocd-server Synced  Healthy   service/argocd-server unchanged
argoproj.io Application argocd    my-app   Synced   application.argoproj.io/my-app configured
PS E:\projects\terraformProj\task9\1\aks-app-deploy> git add .
PS E:\projects\terraformProj\task9\1\aks-app-deploy> git commit -m "Updated deployment and secretproviderclass"
[main 6a3d701] Updated deployment and secretproviderclass
2 files changed, 58 insertions(+)
create mode 100644 deployment.yaml
create mode 100644 k8s-manifests/secretproviderclass.yaml
PS E:\projects\terraformProj\task9\1\aks-app-deploy> git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.19 KiB | 607.00 KiB/s, done.

```

The screenshot shows the ArgoCD UI for the 'my-secrets-provider' resource. The left sidebar shows the ArgoCD logo and navigation links for Applications, Settings, User Info, Documentation, Resource filters, SYNC STATUS, and HEALTH STATUS. The main panel shows the resource details for 'my-secrets-provider'. The top right has a 'SYNC' button and a 'DELETE' button. The 'SUMMARY' tab is active, showing the resource kind as 'SecretProviderClass', name as 'my-secrets-provider', namespace as 'default', creation time as '02/13/2025 21:22:16 (an hour ago)', and status as 'Synced'. The 'EVENTS' tab is also visible. At the bottom, there are tabs for 'LIVE MANIFEST', 'DIFF', and 'DESIRED MANIFEST', with the 'LIVE MANIFEST' tab selected. A note at the bottom says 'apiVersion: secrets-store.csi.x-k8s.io/v1'. There are checkboxes for 'Hide Managed Fields' and 'Enable Word W'.

Monitoring and Logging

Microsoft Azure

Home > Kubernetes services > aks-production

aks-production | Monitor

Kubernetes service

Search ...

Cost analysis

Kubernetes resources

- Namespaces
- Workloads
- Services and ingresses
- Storage
- Configuration
- Custom resources
- Events
- Run command

Settings

Monitoring

- Insights
- Alerts
- Metrics
- Diagnostic settings
- Advisor recommendations
- Logs
- Workbooks

Automation

Refresh Monitor Settings View Grafana Recommended alerts (Preview) View All Clusters

Time range = Last hour

Overview Nodes Workloads Containers

Events Total 37 Warning 5 Normal 32

Alerts Enable recommended alert rules

Get notified on important monitoring events by enabling commonly used alert rules or creating your own custom rules.

Enable Create alert rule

Cluster Summary

Nodes Total 3 Ready (3) Not Ready (0)

Pods Total 43 Failed (0)

Configure Cancel

Configure Azure monitor for containers

Container logs

Container logs collects container logs, Kubernetes events, and cluster inventory data with the Azure Monitor agent to enable debugging your applications. Learn more

Enable container logs

We will onboard your existing Log Analytics workspace (defaultworkspace-9a6ae428-d8c3-44fe-bdf2-4e08593901a0-weu).

Managed Prometheus

Managed Prometheus provides a highly available, scalable and secure metrics platform to monitor your containerized workloads. Learn more

Enable Prometheus metrics

We will onboard your existing Azure Monitor workspace (defaultazuremonitorworkspace-eus).

Managed Grafana

Selecting a fully managed instance of Grafana to visualize your managed Prometheus data stored in your Azure Monitor workspace. Learn more about pricing

Enable Grafana

We will onboard your existing Grafana instance (grafana-202311155652-ne).

Advanced settings

Container logs

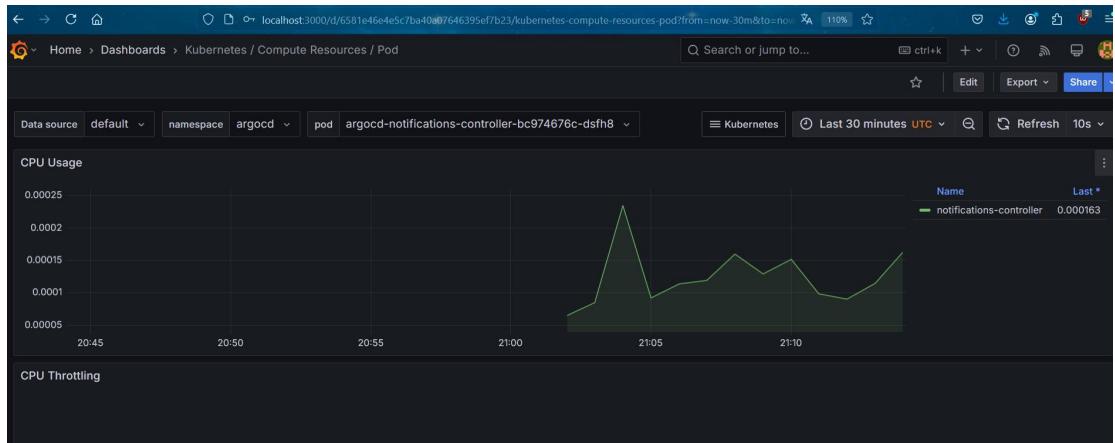
Log Analytics workspace defaultworkspace-9a6ae428-d8c3-44fe-bdf2-4e08593901a0-weu

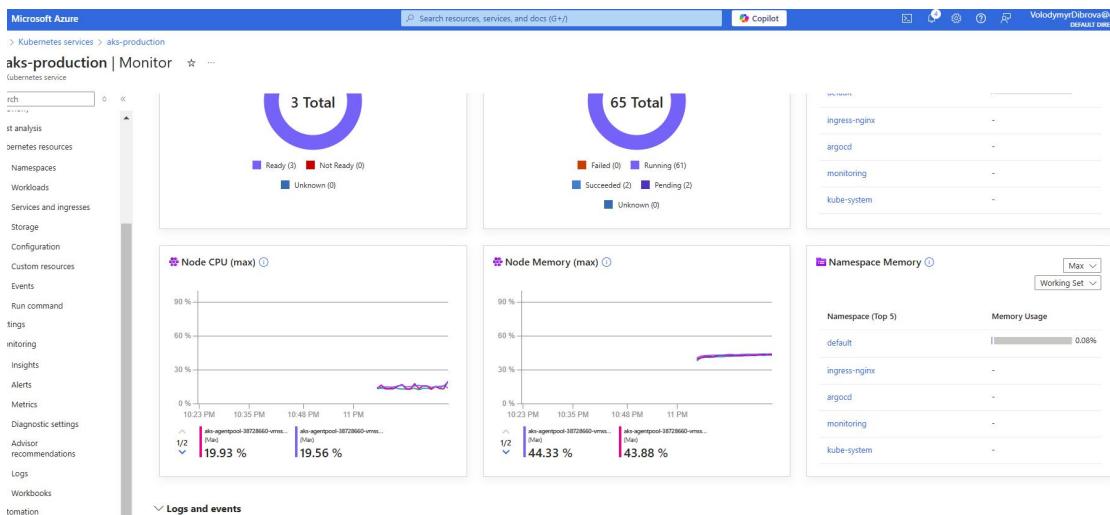
Cost presets Logs and events

Edit collection settings

1m collection frequency

```
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl get svc -n monitoring
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP  PORT(S)        AGE
alertmanager-operated   ClusterIP  None         <none>       9093/TCP,9094/TCP,9094/UDP  8s
prometheus-grafana     ClusterIP  10.0.13.214 <none>       80/TCP          17s
prometheus-kube-prometheus-alertmanager   ClusterIP  10.0.152.41  <none>       9093/TCP,8080/TCP  17s
prometheus-kube-prometheus-operator        ClusterIP  10.0.127.254 <none>       443/TCP          17s
prometheus-kube-prometheus-prometheus    ClusterIP  10.0.61.253  <none>       9090/TCP,8080/TCP  17s
prometheus-kube-state-metrics            ClusterIP  10.0.36.135 <none>       8080/TCP          17s
prometheus-operated             ClusterIP  None         <none>       9090/TCP          8s
prometheus-prometheus-node-exporter     ClusterIP  10.0.201.147 <none>       9100/TCP          17s
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl port-forward svc/prometheus-grafana 3000:80 -n monitoring
error: unable to forward port because pod is not running. Current status=Pending
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl get pods -n monitoring
NAME                           READY   STATUS    RESTARTS   AGE
alertmanager-prometheus-kube-prometheus-alertmanager-0   2/2    Running   0          2m45s
prometheus-grafana-856fb5d8c-pn4d2                      3/3    Running   0          2m12s
prometheus-kube-prometheus-operator-94cc5845d-w6v45       1/1    Running   0          2m12s
prometheus-kube-state-metrics-fff54496d6-vq46l           1/1    Running   0          2m12s
prometheus-prometheus-kube-prometheus-prometheus-0       2/2    Running   0          2m45s
prometheus-prometheus-node-exporter-kzptd                 1/1    Running   0          2m13s
prometheus-prometheus-node-exporter-m8vd8                 1/1    Running   0          2m13s
prometheus-prometheus-node-exporter-mz29k                 1/1    Running   0          2m13s
PS E:\projects\terraformProj\task9\1\aks-app-deploy> kubectl port-forward svc/prometheus-grafana 3000:80 -n monitoring
Forwarding from 127.0.0.1:3000 -> 3000
Forwarding from [::]:3000 -> 3000
Handling connection for 3000
```





Practical Task 10: Deploy a Virtual Machine with Networking Using an ARM Template

Requirements:

Create an ARM template that deploys:

- An **Azure Virtual Network (VNet)** with a **custom subnet**.
- A **Network Security Group (NSG)** allowing SSH and HTTP traffic.
- A **Virtual Machine (VM)** running **Ubuntu 20.04**.
- A **Public IP Address** assigned to the VM.
- Use **parameters** for VM name, admin username, and authentication type.
- Deploy the template and verify:
 - The VM is accessible via SSH.
 - The public IP is assigned correctly.
- Modify the template to enable **boot diagnostics** and redeploy.
- Delete the VM and associated resources after verification.

```
{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "vmName": {
      "type": "string",
      "metadata": {
        "description": "Virtual machine name"
      }
    }
  }
}
```

```
        },
        "adminUsername": {
            "type": "string",
            "metadata": {
                "description": "Administrator username for the virtual machine"
            }
        },
        "adminPassword": {
            "type": "securestring",
            "metadata": {
                "description": "Admin password (if password is used)"
            }
        },
        "authenticationType": {
            "type": "string",
            "defaultValue": "password",
            "allowedValues": [
                "password",
                "sshPublicKey"
            ],
            "metadata": {
                "description": "Type of authentication (password or sshPublicKey)"
            }
        },
        "adminPublicKey": {
            "type": "string",
            "defaultValue": "",
            "metadata": {
                "description": "Public key for SSH (if sshPublicKey is used)"
            }
        },
        "storageAccountName": {
            "type": "string",
            "metadata": {
                "description": "The name of the storage account for boot diagnostics (must be globally unique and in lower case)."
            }
        }
    },
    "variables": {
        "vnetName": "[concat(parameters('vmName'), '-vnet')]",
        "subnetName": "default",
        "nsgName": "[concat(parameters('vmName'), '-nsg')]",
        "ipName": "[concat(parameters('vmName'), '-ip')]",
        "nicName": "[concat(parameters('vmName'), '-nic')]",
        "ubuntuImage": {
            "publisher": "Canonical",
            "offer": "UbuntuServer",
            "sku": "18.04-LTS",
            "version": "latest"
        }
    },
    "resources": [
        {
            "type": "Microsoft.Storage/storageAccounts",
            "apiVersion": "2019-06-01",
            "name": "[parameters('storageAccountName')]",
            "location": "[resourceGroup().location]",
```

```
        "sku": {
            "name": "Standard_LRS"
        },
        "kind": "StorageV2",
        "properties": {}
    },
    {
        "type": "Microsoft.Network/networkSecurityGroups",
        "apiVersion": "2020-06-01",
        "name": "[variables('nsgName')]",
        "location": "[resourceGroup().location]",
        "properties": {
            "securityRules": [
                {
                    "name": "Allow-SSH",
                    "properties": {
                        "protocol": "Tcp",
                        "sourcePortRange": "*",
                        "destinationPortRange": "22",
                        "sourceAddressPrefix": "*",
                        "destinationAddressPrefix": "*",
                        "access": "Allow",
                        "priority": 1000,
                        "direction": "Inbound"
                    }
                },
                {
                    "name": "Allow-HTTP",
                    "properties": {
                        "protocol": "Tcp",
                        "sourcePortRange": "*",
                        "destinationPortRange": "80",
                        "sourceAddressPrefix": "*",
                        "destinationAddressPrefix": "*",
                        "access": "Allow",
                        "priority": 1010,
                        "direction": "Inbound"
                    }
                }
            ]
        }
    },
    {
        "type": "Microsoft.Network/virtualNetworks",
        "apiVersion": "2020-06-01",
        "name": "[variables('vnetName')]",
        "location": "[resourceGroup().location]",
        "dependsOn": [
            "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('nsgName'))]"
        ],
        "properties": {
            "addressSpace": {
                "addressPrefixes": [
                    "10.0.0.0/16"
                ]
            },
            "subnets": [
                {

```

```

        "name": "[variables('subnetName')]",
        "properties": {
            "addressPrefix": "10.0.0.0/24",
            "networkSecurityGroup": {
                "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('nsgName'))]"
            }
        }
    },
    {
        "type": "Microsoft.Network/publicIPAddresses",
        "apiVersion": "2020-06-01",
        "name": "[variables('ipName')]",
        "location": "[resourceGroup().location]",
        "sku": {
            "name": "Basic"
        },
        "properties": {
            "publicIPAllocationMethod": "Dynamic"
        }
    },
    {
        "type": "Microsoft.Network/networkInterfaces",
        "apiVersion": "2020-06-01",
        "name": "[variables('nicName')]",
        "location": "[resourceGroup().location]",
        "dependsOn": [
            "[resourceId('Microsoft.Network/virtualNetworks',
variables('vnetName'))]",
            "[resourceId('Microsoft.Network/publicIPAddresses',
variables('ipName'))]"
        ],
        "properties": {
            "ipConfigurations": [
                {
                    "name": "ipconfig1",
                    "properties": {
                        "subnet": {
                            "id": "[resourceId('Microsoft.Network/subnets',
variables('vnetName')), variables('subnetName'))]"
                        },
                        "privateIPAllocationMethod": "Dynamic",
                        "publicIPAddress": {
                            "id": "[resourceId('Microsoft.Network/publicIPAddresses',
variables('ipName'))]"
                        }
                    }
                }
            ]
        }
    },
    {
        "type": "Microsoft.Compute/virtualMachines",
        "apiVersion": "2021-07-01",

```

```

    "name": "[parameters('vmName')]",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkInterfaces',
variables('nicName'))]",
        "[resourceId('Microsoft.Storage/storageAccounts',
parameters('storageAccountName'))]"
    ],
    "properties": {
        "hardwareProfile": {
            "vmSize": "Standard_B1s"
        },
        "osProfile": {
            "computerName": "[parameters('vmName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "adminPassword": "[if>equals(parameters('authenticationType'),
'password'), parameters('adminPassword'), json('null'))]",
            "linuxConfiguration": {
                "disablePasswordAuthentication":
"[equals(parameters('authenticationType'), 'sshPublicKey')]",
                "ssh": "[if>equals(parameters('authenticationType'),
'sshPublicKey'), json(concat('{\"publicKeys\": [{\"path\": \"/home/',
parameters('adminUsername'), '/.ssh/authorized_keys\", \"keyData\": \"',
parameters('adminPublicKey'), '\"}]}'), json('{\"publicKeys\": []}'))]"
            }
        },
        "storageProfile": {
            "imageReference": {
                "publisher": "[variables('ubuntuImage').publisher]",
                "offer": "[variables('ubuntuImage').offer]",
                "sku": "[variables('ubuntuImage').sku]",
                "version": "[variables('ubuntuImage').version]"
            },
            "osDisk": {
                "createOption": "FromImage"
            }
        },
        "networkProfile": {
            "networkInterfaces": [
                {
                    "id": "[resourceId('Microsoft.Network/networkInterfaces',
variables('nicName'))]"
                }
            ]
        },
        "diagnosticsProfile": {
            "bootDiagnostics": {
                "enabled": true,
                "storageUri": "[concat('https://',
parameters('storageAccountName'), '.blob.core.windows.net/')]"
            }
        }
    }
],
"outputs": {
    "publicIP": {
        "type": "string",

```

```
        "value": "[reference(resourceId('Microsoft.Network/publicIPAddresses',  
variables('ipName')), '2020-06-01'). ipAddress]"  
    }  
}  
}
```

```
PS E:\projects\terraformProj\task10> az deployment group create '  
>> --resource-group Volodymyr-Dibrova '  
>> --template-file azuredeploy.json '  
>> --parameters vmName=myVM adminUsername=azureuser adminPassword="myPassword123!" authenticationType=password  
{  
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Resources/deployments/azureddeploy",  
  "location": null,  
  "name": "azureddeploy",  
  "properties": {  
    "correlationId": "e68d5d5e-5d10-473c-b917-26ec157c72fe",  
    "debugSetting": null,  
    "dependencies": [  
      {  
        "dependsOn": [  
          {  
            "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/networkSecurityGroups/myVM-nsg",  
            "resourceGroup": "Volodymyr-Dibrova",  
            "resourceName": "myVM-nsg",  
            "resourceType": "Microsoft.Network/networkSecurityGroups"  
          },  
          {  
            "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/myVM-vnet",  
            "resourceGroup": "Volodymyr-Dibrova",  
            "resourceName": "myVM-vnet",  
            "resourceType": "Microsoft.Network/virtualNetworks"  
          },  
          {  
            "dependsOn": [  
              {  
                "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/virtualNetworks/myVM-vnet",  
                "resourceGroup": "Volodymyr-Dibrova",  
                "resourceName": "myVM-vnet",  
                "resourceType": "Microsoft.Network/virtualNetworks"  
              },  
              {  
                "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/publicIPAddresses/myVM-i  
t",  
                "resourceGroup": "Volodymyr-Dibrova",  
                "resourceName": "myVM-ip",  
                "resourceType": "Microsoft.Network/publicIPAddresses"  
              }  
            ]  
          }  
        ]  
      }  
    ]  
  }  
}
```

```

az user@myVM: ~          + - 
"templateHash": "17160973442003344708",
"templateLink": null,
"timestamp": "2025-02-14T20:34:44.657057+00:00",
"validatedResources": null
},
"resourceGroup": "Volodymyr-Dibrova",
"tags": null,
"type": "Microsoft.Resources/deployments"
}
PS E:\projects\terraformProj\task10> ssh azureuser@52.232.29.27
azureuser@52.232.29.27's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1109-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Fri Feb 14 20:46:21 UTC 2025

 System load:  0.0          Processes:      104
 Usage of /:   4.6% of 28.89GB   Users logged in:   0
 Memory usage: 21%           IP address for eth0: 10.0.0.4
 Swap usage:   0%

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Fri Feb 14 20:17:32 2025 from 95.158.42.247
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@myVM:~$ |

```

```

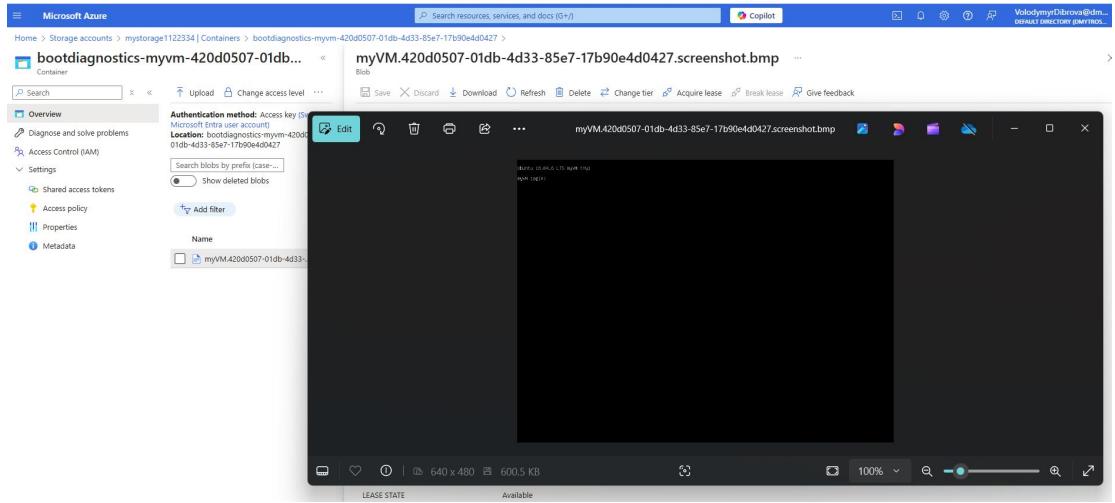
PS E:\projects\terraformProj\task10> az network public-ip show --resource-group Volodymyr-Dibrova --name myVM-ip --query ipAddress --output tsv
52.232.29.27
PS E:\projects\terraformProj\task10> |

```

```

206 },
207     "networkProfile": {
208         "networkInterfaces": [
209             {
210                 "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]"
211             }
212         ]
213     },
214     "diagnosticsProfile": {
215         "bootDiagnostics": {
216             "enabled": true,
217             "storageUri": "[concat('https://', parameters('storageAccountName'), '.blob.core.windows.net/'])"
218         }
219     }
220 },
221 ],
222 "outputs": [
223     {
224         "publicIP": {

```



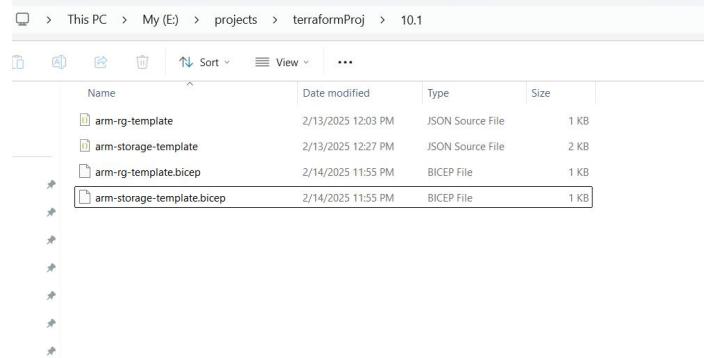
10.1 Azure Bicep Tasks

Convert an ARM Template to Bicep

Requirements:

Take an existing **ARM Template** (e.g., from Task 8) and convert it to **Azure Bicep** format using:

- **az bicep decompile --file <arm-template.json>**
- Refactor the Bicep template to:
 - Remove unnecessary metadata.
 - Use **variables** instead of hardcoded values.
 - Use **parameters** for the storage account name, SKU, and location.
- Deploy the Bicep file using **az deployment group create**.
- Validate the deployment and compare it to the original ARM deployment.
- Delete the storage account after testing.



```

1 targetScope = 'subscription'
2
3 @description('The name of the resource group.')
4 param resourceName string = 'ARMResourceGroup'
5
6 @description('Location for the resource group.')
7 param location string = 'eastus'
8
9 var tags = {
10   Environment: 'Development'
11   Owner: 'Volodymyr Dibrova'
12   Project: 'Terraform Task 8'
13 }
14
15 resource resourceGroup 'Microsoft.Resources/resourceGroups@2021-04-01' = {
16   name: resourceName
17   location: location
18   properties: {}
19   tags: tags
20 }
21

```

```

1 @description('The name of the storage account (must be globally unique and in lower case).')
2 param storageAccountName string
3
4 @description('The location where the storage account will be deployed.')
5 param storageLocation string = resourceGroup().location
6
7 @description('The SKU of the storage account.')
8 param storageSku string = 'Standard_LRS'
9
10 resource storageAccount 'Microsoft.Storage/storageAccounts@2022-09-01' = {
11   name: storageAccountName
12   location: storageLocation
13   kind: 'StorageV2'
14   sku: {
15     name: storageSku
16   }
17   properties: {
18     accessTier: 'Hot'
19     allowBlobPublicAccess: false
20     deleteRetentionPolicy: {
21       enabled: true
22       days: 7
23     }
24   }
25 }
26
27 output storageAccountId string = storageAccount.id
28

```

```

PS E:\projects\terraformProj\10.1> az deployment sub create \
>> --location eastus \
>> --template-file arm-rg-template.bicep \
>> --parameters resourceName="Volodymyr-Dibrova"
{
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/providers/Microsoft.Resources/deployments/arm-rg-template",
  "location": "eastus",
  "name": "arm-rg-template",
  "properties": {
    "correlationId": "384df618-ed8e-46ea-8f57-0b30dd6695af",
    "debugSetting": null,
    "dependencies": [],
    "duration": "PT4.00918865",
    "error": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova"
      }
    ],
    "outputs": null,
    "parameters": {
      "location": {
        "type": "String",
        "value": "eastus"
      }
    }
  }
}

```

```

PS E:\projects\terraformProj\10.1> az deployment group create ` 
>>   --resource-group "Volodymyr-Dibrova" ` 
>>   --template-file arm-storage-template.bicep ` 
>>   --parameters storageAccountName="mystorageuniqueusername123" storageLocation="eastus" storageSku="Standard_LRS" 
E:\projects\terraformProj\10.1\arm-storage-template.bicep(20,5) : Warning BCP037: The property "deleteRetentionPolicy" is not allowed on objects of type "StorageAccountPropertiesCreateParametersOrStorageAccountProperties". Permissible properties include "allowCrossTenantReplication", "allowedCopyScope", "allowSharedKeyAccess", "azureFilesIdentityBasedAuthentication", "customDomain", "defaultToOAuthAuthentication", "dnsEndpointType", "encryption", "immutableStorageWithVersioning", "isHnsEnabled", "isLocalUserEnabled", "isNfsV3Enabled", "isSftpEnabled", "keyPolicy", "largeFileSharesState", "minimumTlsVersion", "networkAcls", "publicNetworkAccess", "routingPreference", "sasPolicy", "supportsHttpsTrafficOnly". If this is a resource type definition inaccuracy, report it using https://aka.ms/bicep-type-issues. [https://aka.ms/bicep/re-diagnostics#BCP037] 

{
  "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Resources/deployments/arm-storage-template",
  "location": null,
  "name": "arm-storage-template",
  "properties": {
    "correlationId": "83bd4aab-013b-43c1-b9df-06612292b325",
    "debugSetting": null,
    "dependencies": [],
    "duration": "PT24.007339S",
    "error": null,
    "mode": "Incremental",
    "onErrorDeployment": null,
    "outputResources": [
      {
        "dependsOn": [
          "Microsoft.Storage/storageAccounts/mystorageuniqueusername123"
        ],
        "name": "mystorageuniqueusername123",
        "properties": {
          "accountType": "Standard_LRS",
          "blobContainerSku": "Standard_RRS",
          "deleteRetentionPolicy": {
            "enabled": true,
            "retentionDays": 30
          },
          "encryption": {
            "keyEncryptionType": "AccountKey"
          },
          "geoReplicationPrimaryLocation": "East US",
          "geoReplicationSecondaryLocation": "West US",
          "httpsPort": true,
          "minTLSVersion": "TLS1_2"
        }
      }
    ]
  }
}

```

The screenshot shows the Azure portal interface for the 'Volodymyr-Dibrova' resource group. The 'Overview' tab is selected. In the 'Resources' section, there is one item listed: a Storage account named 'mystorageuniqueusername123'. The 'Essentials' panel provides details about the subscription (Subscription ID: 9a6ae428-d8c3-44fe-bdf2-4e08593901a0, Location: East US, Deployments: 1 Succeeded) and the resource itself (Tags: Environment: Development, Owner: Volodymyr Dibrova, Project: Terraform Task 8).

Practical Task 11: Deploy a Multi-Resource Azure Infrastructure Using Bicep Requirements:

Create an **Azure Bicep** file that deploys:

- A **Virtual Network (VNet)** with multiple **subnets**.
- A **Storage Account** for VM diagnostics.
- A **Linux Virtual Machine** with SSH authentication.
- A **Network Security Group (NSG)** with restricted SSH access.
- Implement **modules** in Bicep for:
 - Networking
 - Virtual Machine Deployment
- Deploy the Bicep file and verify:
 - The VM is reachable via SSH.
 - The NSG allows the correct traffic.
- Modify the configuration to increase VM size and redeploy.
- Destroy the infrastructure after verification.

Created vm by bicep

The screenshot shows a code editor interface with a dark theme. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'task11'. Below the navigation bar is a tab bar with tabs for Welcome, network.bicep, vm.bicep (which is currently active), and main.bicep. The status bar at the bottom right shows 'Extension: Bicep'.

The left sidebar has sections for EXPLORER, TASK11, and OUTLINE. The OUTLINE section is expanded, showing a tree structure with nodes like 'vm.bicep > vm', 'networkProfile', 'storageProfile', and 'diagnosticsProfile'.

The main content area displays the Bicep code for creating a virtual machine:

```
param vmName string = 'mylinuxVM'
param location string = resourceGroup().location
param adminUsername string = 'azureuser'
param sshPublicKey string
param vmSize string = 'Standard_B1s'
param subnetId string
param diagnosticsStorageAccountId string

resource vm 'Microsoft.Compute/virtualMachines@2021-07-01' = {
    name: vmName
    location: location
    properties: {
        hardwareProfile: {
            vmSize: vmSize
        }
        osProfile: []
        computerName: vmName
        adminUsername: adminUsername
        linuxConfiguration: {
            disablePasswordAuthentication: true
            ssh: {
                publicKeys: [
                    {
                        path: '/home/${adminUsername}/.ssh/authorized_keys'
                        keyData: sshPublicKey
                    }
                ]
            }
        }
        networkProfile: {
            networkInterfaces: [
                {
                    id: subnetId
                }
            ]
        }
        diagnosticsProfile: {
            bootDiagnostics: {
                enabled: true
                storageUri: reference(diagnosticsStorageAccountId, '2021-04-01').primaryEndpoints.blob
            }
        }
        storageProfile: {
            imageReference: {
                publisher: 'Canonical'
                offer: 'UbuntuServer'
                sku: '18.04-LTS'
                version: 'latest'
            }
            osDisk: {
                createOption: 'FromImage'
            }
        }
    }
}

output vmId string = vm.id
```

```
... network.bicep ● vm.bicep main.bicep my.py main.py  
network.bicep > {} vnet  
1 param vnetName string = 'myVNet'  
2 param location string = resourceGroup().location  
3 param addressPrefix string = '10.0.0.0/16'  
4 param subnetPrefix1 string = '10.0.1.0/24'  
5 param subnetPrefix2 string = '10.0.2.0/24'  
6 param allowedIP string = 'xx.xx.xx.xx/32'  
7 resource vnet 'Microsoft.Network/virtualNetworks@2021-05-01' = {  
8   name: vnetName  
9   location: location  
10  properties: []  
11    addressSpace: {  
12      addressPrefixes: [  
13        addressPrefix  
14      ]  
15    }  
16    subnets: [  
17      {  
18        name: 'subnet1'  
19        properties: {  
20          addressPrefix: subnetPrefix1  
21        }  
22      }  
23      {  
24        name: 'subnet2'  
25        properties: {  
26          addressPrefix: subnetPrefix2  
27        }  
28      }  
29    ]  
30  ]  
31 }  
32  
33 resource nsg 'Microsoft.Network/networkSecurityGroups@2021-02-01' = {  
34   name: '${vnetName}-nsg'  
35   location: location  
36   properties: {  
37     securityRules: [  
38       {  
39         name: 'Allow-SSH'  
40         properties: {  
41           protocol: 'Tcp'  
42           sourcePortRange: '*'  
43           destinationPortRange: '22'  
44           sourceAddressPrefix: allowedIP  
45           destinationAddressPrefix: '*'  
46           access: 'Allow'  
47           priority: 1000  
48           direction: 'Inbound'  
49         }  
50       }  
51     ]  
52   }  
53 }  
54  
55 output subnet1ID string = resourceId('Microsoft.Network/virtualNetworks/subnets', vnet.name, 'subnet1')  
56 output nsgId string = nsg.id  
57
```

```
32  
33 resource nsg 'Microsoft.Network/networkSecurityGroups@2021-02-01' = {  
34   name: '${vnetName}-nsg'  
35   location: location  
36   properties: {  
37     securityRules: [  
38       {  
39         name: 'Allow-SSH'  
40         properties: {  
41           protocol: 'Tcp'  
42           sourcePortRange: '*'  
43           destinationPortRange: '22'  
44           sourceAddressPrefix: allowedIP  
45           destinationAddressPrefix: '*'  
46           access: 'Allow'  
47           priority: 1000  
48           direction: 'Inbound'  
49         }  
50       }  
51     ]  
52   }  
53 }  
54  
55 output subnet1ID string = resourceId('Microsoft.Network/virtualNetworks/subnets', vnet.name, 'subnet1')  
56 output nsgId string = nsg.id  
57
```

Main.bicep

```
param location string = resourceGroup().location
param adminUsername string = 'azureuser'
param sshPublicKey string
param vmSize string = 'Standard_DS2_v2'
param allowedIP string = '203.0.113.45/32'

resource diagnosticsStorage 'Microsoft.Storage/storageAccounts@2021-04-01' =
{
    name: 'diag${uniqueString(resourceGroup().id)}'
    location: location
    sku: {
        name: 'Standard_LRS'
    }
    kind: 'Storage'
    properties: {}
}

module networkModule './network.bicep' = {
    name: 'networkModule'
    params: {
        vnetName: 'myVNet'
        location: location
        allowedIP: allowedIP
    }
}

resource nic 'Microsoft.Network/networkInterfaces@2021-05-01' = {
    name: 'myNIC'
    location: location
    properties: {
        ipConfigurations: [
            {
                name: 'ipconfig1'
                properties: {
                    subnet: {
                        id: networkModule.outputs.subnet1Id
                    }
                    privateIPAllocationMethod: 'Dynamic'
                }
            }
        ]
        networkSecurityGroup: {
            id: networkModule.outputs.nsgId
        }
    }
}

// Module vm
module vmModule './vm.bicep' = {
    name: 'vmModule'
    params: {
        vmName: 'myLinuxVM'
        location: location
        adminUsername: adminUsername
        sshPublicKey: sshPublicKey
        vmSize: vmSize
    }
}
```

```

        subnetId: nic.id
        diagnosticsStorageAccountId: diagnosticsStorage.id
    }
}

output vmId string = vmModule.outputs.vmId

```

```

PS E:\projects\terraformProj\task11> az deployment group create --resource-group Volodymyr-Dibrova --template-file main.bicep
Please provide string value for 'sshPublicKey' (?) for help: ssh-rsa AAAAB3NzaC1yc2EAAQAAQAAAgQDIDaB/St+1kR4xUW0uUf5Vzx8otkj6kBNL9k1bi1xsbsXjM+0tkU0kJ3
tTDKc6LMqmwooolublii/b2fxttcovlU3018mcBHTipjh4fVZBV5FRW/3G1nTb0/gvNTH/wbp9gWa7mJe18bZgU62i3L5q1NtdBdemLNp8V0p9u5axx2RZ0Lacv96pagJYD44qan0dcPteIxy0p
CdANhDBGU/SUST+9itFGqv+1jt+LwmbuXuQP03Xn+g1OIITP90ATVm+ap/W8NPJZvvgCcFu4zu/CuNNl7UqJLGrxDyusNm33o2A6vTt+BkOP9byzitvC4Bb64X9rhG48nVdZ7phprHECp8YXRnHhc
TBAGyIYkddUyche0fSS2JkFahlBBVDHeU8iETD0RHE2KCXYQvcUmbTEdxmH5H9Ce42tBovkMkgolfvfLELUc1h0p0US27cDRejFwHIhu01/ALRPAN2MYCZ7gIHnJ4pcQGwb5qCMMuCNC= infinite
m24@gmail.com
{
    "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Resources/deployments/main",
    "location": null,
    "name": "main",
    "properties": {
        "correlationId": "057565c4-7b5f-49a1-b603-de77930be498",
        "debugSetting": null,
        "dependencies": [
            {
                "dependsOn": [
                    {
                        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Resources/deployments/networkMod
ule",
                        "resourceGroup": "Volodymyr-Dibrova",
                        "resourceName": "networkModule",
                        "resourceType": "Microsoft.Resources/deployments"
                    },
                    {
                        "apiVersion": "2022-09-01",
                        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Resources/deployments/networkMod
ule",
                        "resourceGroup": "Volodymyr-Dibrova",
                        "resourceName": "networkModule",
                        "resourceType": "Microsoft.Resources/deployments"
                    }
                ],
                "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Network/networkInterfaces/myNIC",
                "resourceGroup": "Volodymyr-Dibrova",
                "resourceName": "myNIC",
                "resourceType": "Microsoft.Network/networkInterfaces"
            },
            {
                "dependsOn": [
                    {
                        "id": "/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Storage/storageAccounts/diags3yd

```

```

PS E:\projects\terraformProj\task11> az vm list-ip-addresses --resource-group Volodymyr-Dibrova --name myLinuxVM --query "[0].virtualMachine.network.publicIpAddresses[0]. ipAddress" -o tsv
PS E:\projects\terraformProj\task11> az network public-ip list --resource-group Volodymyr-Dibrova -o table
Name      ResourceGroup      Location      Zones      IdleTimeoutInMinutes      ProvisioningState
-----  -----
myVM-ip   Volodymyr-Dibrova   eastus       4          Succeeded
PS E:\projects\terraformProj\task11> az vm show --resource-group Volodymyr-Dibrova --name myLinuxVM --query "hardwarePro
file.vmSize" -o tsv
Standard_B1s
PS E:\projects\terraformProj\task11>

```

```

        "resourceType": "deployments",
        "zoneMappings": null
    }
}
],
"provisioningState": "Succeeded",
"templateHash": "16758539367936450541",
"templateLink": null,
"timestamp": "2025-02-15T13:23:07.667171+00:00",
"validatedResources": null
},
"resourceGroup": "Volodymyr-Dibrova",
"tags": null,
"type": "Microsoft.Resources/deployments"
}
PS E:\projects\terraformProj\task11> az vm list-ip-addresses --resource-group Volodymyr-Dibrova --name myLinuxVM --query "[0].virtualMachine.network.publicIpAddresses[0].ipAddress" -o tsv
PS E:\projects\terraformProj\task11> az network public-ip list --resource-group Volodymyr-Dibrova -o table
Name      ResourceGroup      Location      Zones      IdleTimeoutInMinutes      ProvisioningState
-----  -----
myVM-ip   Volodymyr-Dibrova   eastus       4          Succeeded
PS E:\projects\terraformProj\task11> az vm show --resource-group Volodymyr-Dibrova --name myLinuxVM --query "hardwarePro
file.vmSize" -o tsv
Standard_B1s
PS E:\projects\terraformProj\task11>

```

Azure Monitor Tasks

Practical Task 13: Configure Azure Monitor to Track VM Metrics and Alerts

Requirements:

- Enable **Azure Monitor** for a Virtual Machine (VM).
- Configure monitoring for the following **metrics**:
 - CPU utilization
 - Disk read/write operations
 - Network In/Out
- Set up an **alert rule** that triggers when CPU utilization exceeds **80% for 5 minutes**.
- Configure the alert to send a **notification to an email address**.
- Verify the alert trigger by running a CPU-intensive process on the VM.
- Delete the alert rule after testing.

Created Log Analytics workspace

Created metric alert rule for Percentage CPU and added notification email

CPU-Usage-Alert Metric alert rule

Overview

- Resource group (move) : Volodymyr-Dibrova
- Location (move) : Global
- Subscription (move) : Azure subscription 1
- Subscription ID : 9a8ae428-d8c3-4afe-bdf2-4e06593901a0
- Tags (edit) : Add tags

Scope

- Resource : mylinuxvm
- Hierarchy : Azure subscription 1 > [Volodymyr-Dibrova]

Actions

Name	Contains actions
CPU-Alert-Group	1 Email

Conditions

Name	Time series monitored	Estimated monthly cost
Percentage CPU > 80	1	\$0.10

Returned the 10 most recent system performance records

New Query 1

Run (Time range: Last 24 hours)

Results

TimeGenerated [UTC]	Computer	ObjectName	CounterName	InstanceName	CounterValue	CounterPath	Type	ResourceID
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Used Space	total	3.39308365021192	\myLinuxVM\Logical Disk\total%\Used Space	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	Free Megabytes	total	56139691520	\myLinuxVM\Logical Disk\total%\Free Megabytes	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Free Space	total	95.2745786764592	\myLinuxVM\Logical Disk\total%\Free Space	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Used Inodes	/run/user/1000	0.00134888750503023	\myLinuxVM\Logical Disk\run\user\1000%\Used Inodes	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	Free Megabytes	/run/user/1000	695.015625	\myLinuxVM\Logical Disk\run\user\1000%\Free Megabytes	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Free Inodes	total	99.3604692665039	\myLinuxVM\Logical Disk\total%\Free Inodes	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Used Space	/run/user/1000	0	\myLinuxVM\Logical Disk\run\user\1000%\Used Space	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Used Inodes	total	0.639530733494108	\myLinuxVM\Logical Disk\total%\Used Inodes	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Free Space	/run/user/1000	100	\myLinuxVM\Logical Disk\run\user\1000%\Free Space	Perf	/subscriptions/
> 2/15/2025, 5:37:00:00 PM	mylinuxVM	Logical Disk	% Free Inodes	/run/user/1000	99.99865112495	\myLinuxVM\Logical Disk\run\user\1000%\Free Inodes	Perf	/subscriptions/

Notification email

Action groups

CPU-Alert-Group Action group

Overview

- Resource group (move) : Volodymyr-Dibrova
- Location (move) : Global
- Subscription (move) : Azure subscription 1
- Subscription ID : 9a8ae428-d8c3-4afe-bdf2-4e06593901a0
- Tags (edit) : Add tags

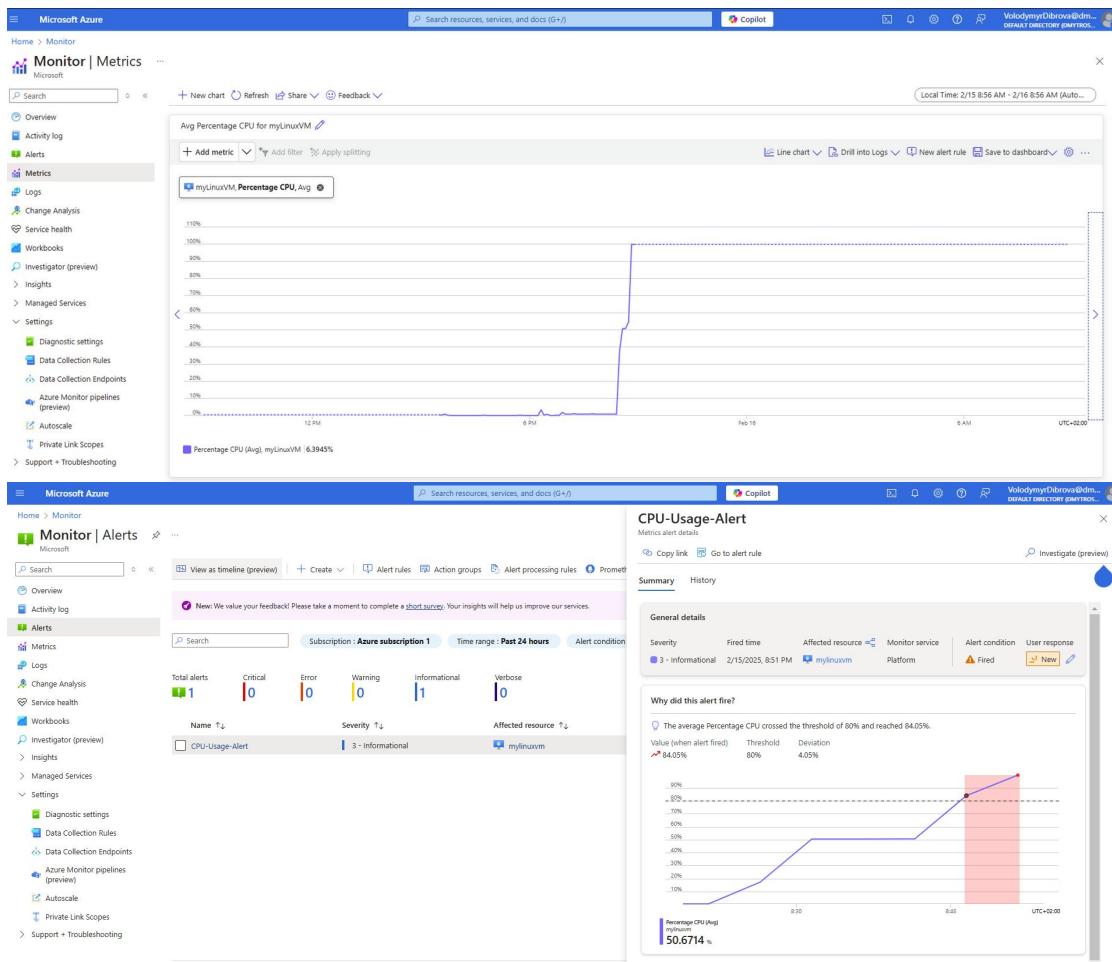
Notifications

Type	Name	Value
Email	email1	VolodymyrDibrova@gmail.com...

```

8 root      0 -20      0      0      0 I    0.0  0.0   0:00.00 mm_percpu_wq
9 root      20  0      0      0      0 S    0.0  0.0   0:00.22 ksoftirqd/0
10 root     20  0      0      0      0 I    0.0  0.0   0:01.09 rcu_sched
11 root     rt  0      0      0      0 S    0.0  0.0   0:00.11 migration/0
13 root     20  0      0      0      0 S    0.0  0.0   0:00.00 cpuhp/0
14 root     20  0      0      0      0 S    0.0  0.0   0:00.00 cpuhp/1
15 root     rt  0      0      0      0 S    0.0  0.0   0:00.60 migration/1
16 root     20  0      0      0      0 S    0.0  0.0   0:00.36 ksoftirqd/1
18 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 kworker/1:0H-kb
19 root     20  0      0      0      0 S    0.0  0.0   0:00.00 kdevtmpfs
20 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 netns
21 root     20  0      0      0      0 S    0.0  0.0   0:00.00 rcu_tasks_kthre
22 root     20  0      0      0      0 S    0.0  0.0   0:00.00 kauditd
23 root     20  0      0      0      0 S    0.0  0.0   0:00.00 khungtaskd
24 root     20  0      0      0      0 S    0.0  0.0   0:00.00 oom_reaper
25 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 writeback
26 root     20  0      0      0      0 S    0.0  0.0   0:00.00 kcompactd0
27 root     25  5      0      0      0 S    0.0  0.0   0:00.00 ksmd
28 root     39  19     0      0      0 S    0.0  0.0   0:00.13 khugepaged
74 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 kintegrityd
75 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 kblockd
76 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 blkcg_punt_bio
77 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 tpm_dev_wq
78 root     0 -20      0      0      0 I    0.0  0.0   0:00.00 ata_sff
azureuser@myLinuxVM:~$ ^C
azureuser@myLinuxVM:~$ yes > /dev/null & yes > /dev/null & yes > /dev/null & yes > /dev/null &
[5] 17871
[6] 17872
[7] 17873
[8] 17874
azureuser@myLinuxVM:~$ yes > /dev/null & yes > /dev/null & yes > /dev/null & yes > /dev/null &
[9] 17963
[10] 17964
[11] 17965
[12] 17966
azureuser@myLinuxVM:~$ |

```



Microsoft Azure

e > Monitor | Data Collection Rules > LinuxVM-Monitoring

LinuxVM-Monitoring | Data sources

Search resources, services, and docs (G+)

Add Delete

View Log Filter by name... D... : Azure Monitor Logs, Azure Monitor Metrics (preview), Az...

Data collection rule

Logs Data source Performance Counters

Activity log Data sources

Logs Automation Policies (Preview) CLI / PS Tasks Export template Security Monitoring

Logs Help

Add data source

*Data source Destination

Select which data source type and the data to collect for your resource(s).

Data source type * Performance Counters

Choose Basic to enable the collection of performance counters. Choose Custom if you want more control over which performance counters are collected.

None Basic Custom

Configure the performance counters to collect, and how often they should be sampled:

Logical Disk

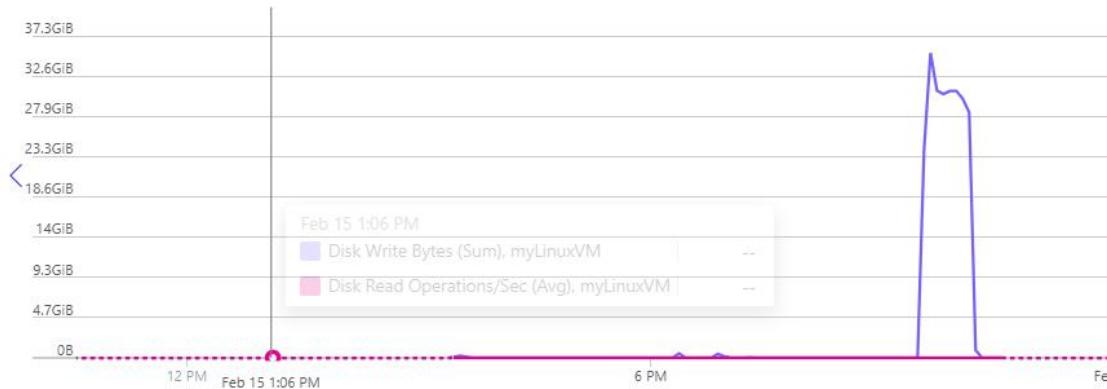
Performance counter

Logical Disk\Logical Disk Bytes/sec Logical Disk Read Bytes/sec Logical Disk Write Bytes/sec

Sample rate (seconds) 60 60 60

Showing 1 - 3 of 3 results. Showing filtered results.

Save Next : Destination > Cancel



Disk Write Bytes (Sum), myLinuxVM | -- Disk Read Operations/Sec (Avg), myLinuxVM | --

Microsoft Azure

Home > Monitor

Monitor | Metrics

Search resources, services, and docs (G+)

New chart Refresh Share Feedback

Local Time: 2/15 9:28 AM - 2/16 9:28 AM (Auto...)

Sum Network In Total and Sum Network Out Total for myLinuxVM

Add metric Add filter Apply splitting

Line chart Drill into Logs New alert rule Save to dashboard ...

myLinuxVM, Network In Total, Sum myLinuxVM, Network Out Total, Sum

286.1MiB 238.4MiB 190.7MiB 143.1MiB 95.4MiB 47.7MiB 0B

12 PM 6 PM Feb 16 6 AM UTC+0200

Network In Total (Sum), myLinuxVM 479.4MiB Network Out Total (Sum), myLinuxVM 34.6MiB

Practical Task 15: Enable Application Insights for a Web Application

Requirements:

- Enable Application Insights for an existing Azure Web App.
- Configure automatic instrumentation for the application (for .NET, Node.js, or Python).
- Monitor the following performance metrics:
 - Response time
 - Request count
 - Failed request rate
- Generate load on the application using Apache JMeter or a similar tool.
- View the performance metrics in Application Insights.
- Set up an alert for high response times (above 2 seconds).

Created WebApp

The screenshot shows the Microsoft Azure portal interface. In the top navigation bar, 'Microsoft Azure' is selected. Below it, 'App Services' and 'Default Directory' are shown. On the left, a sidebar lists several resources under 'App Services': 'mywebapp-1739707561', 'myWebAppDibrova', and another unnamed resource. The main content area is titled 'myWebAppDibrova' and shows the 'Overview' tab. It displays basic information about the web app, such as its name ('myWebAppDibrova'), status ('Running'), location ('West Europe'), and subscription details. A 'Tags' section is also present. Below the overview, there are tabs for 'Properties', 'Monitoring', 'Logs', 'Capabilities', 'Notifications', and 'Recommendations'. The 'Web app' properties are detailed below:

Property	Value
Name	myWebAppDibrova
Publishing model	Code
Runtime Stack	Python - 3.12

Tracing queries in Azure using OpenCensus

The screenshot shows a code editor window with a dark theme. The title bar says 'task11'. The code editor displays a Python file named 'main.py'. The code imports various OpenCensus and Azure modules and sets up an exporter and tracer. The code is as follows:

```
from opencensus.ext.azure.trace_exporter import AzureExporter
from opencensus.trace.tracer import Tracer
from opencensus.trace.samplers import ProbabilitySampler
from opencensus.ext.azure.common.storage import LocalFileStorage
import LocalFileStorage(path='./storage')

exporter = AzureExporter(
    connection_string="InstrumentationKey=b175456c-ffbd-46d9-8e6b-33a87f8b72ef"
)
tracer = Tracer(exporter=exporter, sampler=ProbabilitySampler(1.0))
```

Creating an OpenCensus trace span for client requests

```

my.py > ...
1 import time
2
3 from main import tracer
4 from opencensus.trace.span import SpanKind
5
6 print("Creating test-span...")
7
8 time.sleep(3)
9
10 with tracer.span(name="test-span") as span:
11     span.span_kind = SpanKind.CLIENT
12     span.add_attribute("custom_attribute", "test_value")
13 print("test-span created!")
14
15

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Results from span

Home > myAppDibrova

myAppDibrova | Logs

Application Insights

New Query 1*

Run Time range: Last 24 hours

1 dependencies
2 where customDimensions["custom_attribute"] == "test_value"
3 order by timestamp desc

Results Chart

timestamp [UTC]	id	target	type	name	success
> 2/16/2025, 1:15:52.705 PM	2d982fb5261cb6a9e9	test-span	Other	test-span	True
> 2/16/2025, 1:15:56.448 PM	163474690404f0ff	test-span	Other	test-span	True
> 2/16/2025, 1:16:04.426 PM	e93720d01b6a3be01	test-span	Other	test-span	True
> 2/16/2025, 1:19:45.748 PM	d43720d02798ec	test-span	Other	test-span	True
> 2/16/2025, 1:08:20.669 PM	b1286a59909ff8c7	test-span	Other	test-span	True
> 2/16/2025, 1:08:05.148 PM	7a711ed7fe0984c	test-span	Other	test-span	True
> 2/16/2025, 1:01:30.960 PM	6823324f5a2e90	test-span	Other	test-span	True
> 2/16/2025, 1:25:56.17976 PM	6b26522f3d9bd1da	test-span	Other	test-span	True
> 2/16/2025, 1:25:56.38.249 PM	6a8550f178429466	test-span	Other	test-span	True

Try the new Log An... Feedback Queries hub Pin to Format query

Overview Activity log Access control (IAM) Tags Diagnose and solve problems > Investigate < Monitoring Alerts Metrics Diagnostic settings < Logs Workbooks Usage Configure Properties Smart detection settings Network Isolation

myAppDibrova | Logs

application insights

New Query 1*

Run Time range: Last 24 hours

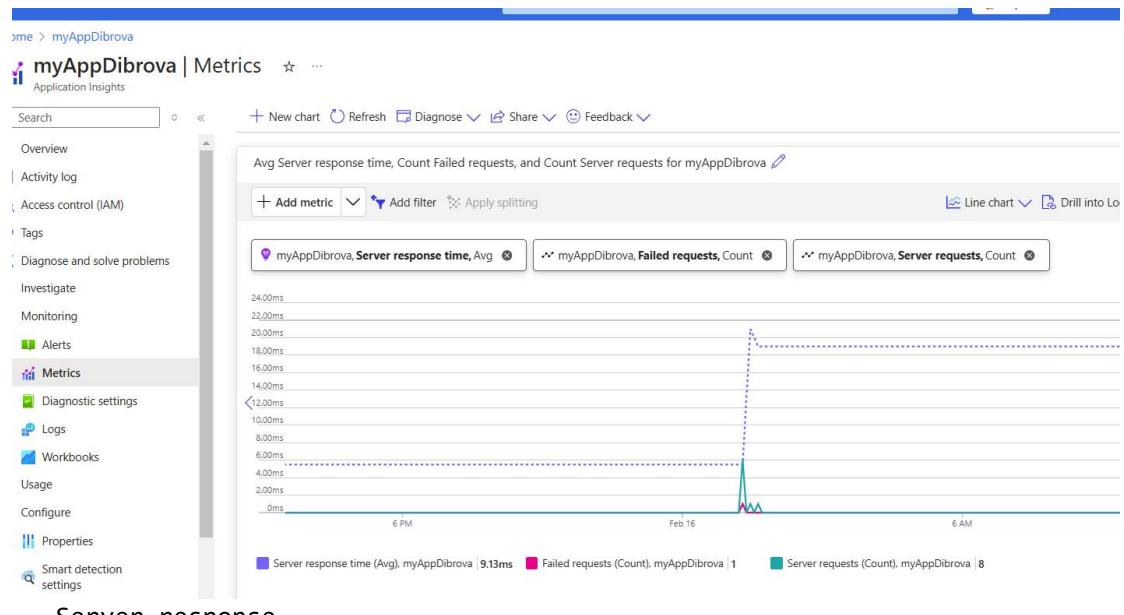
1 traces
2 | order by timestamp desc

Results Chart

timestamp [UTC]	message	severityLevel	itemType	customDimensions	operation_Id
> 2/16/2025, 2:35:26.746 PM	Code Profiler Installer is exiting...	0	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.746 PM	Signal Handlers SIGUSR for ne...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.713 PM	Successfully installed code prof...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.712 PM	viztracer would save traces to /...	0	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.692 PM	Attempting to install the default...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.642 PM	APPSETTING_WEBSITE_ENABLE...	0	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.241 PM	successfully deleted the status f...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.231 PM	Attempting to delete the signal...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.211 PM	Cleaning up any existing status ...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.192 PM	Code Profiler Installer is startin...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	
> 2/16/2025, 2:35:26.192 PM	Code Profiler Installer is startin...	1	trace	{"code.filePath": "/opt/python/3..."} 00000000000000000000000000000000	

Try the new Log An... Feedback Queries hub Pin to Format query

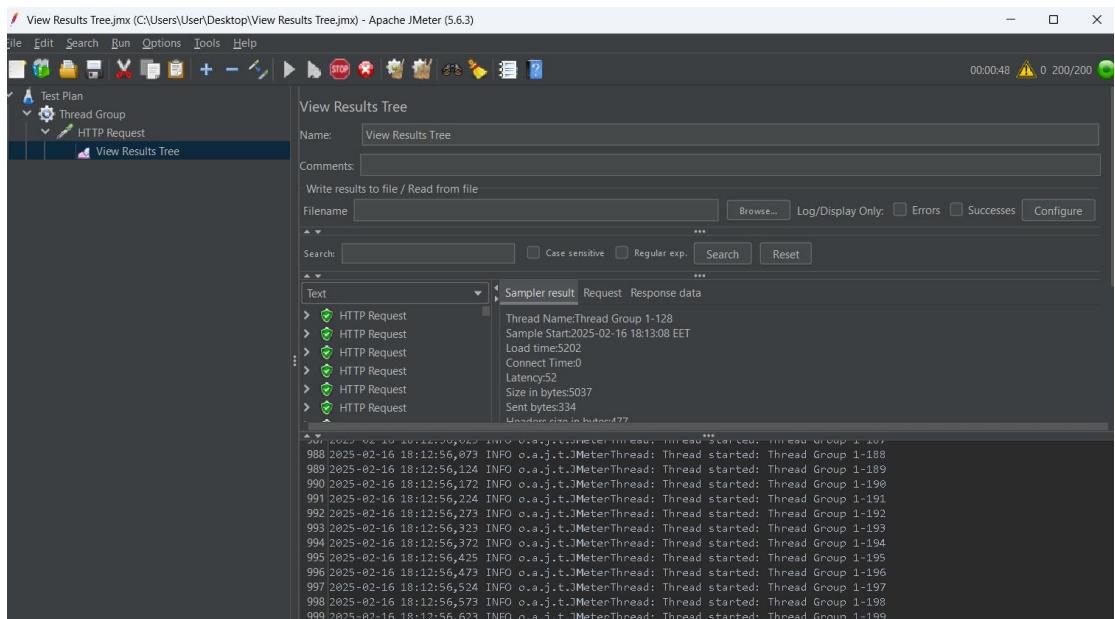
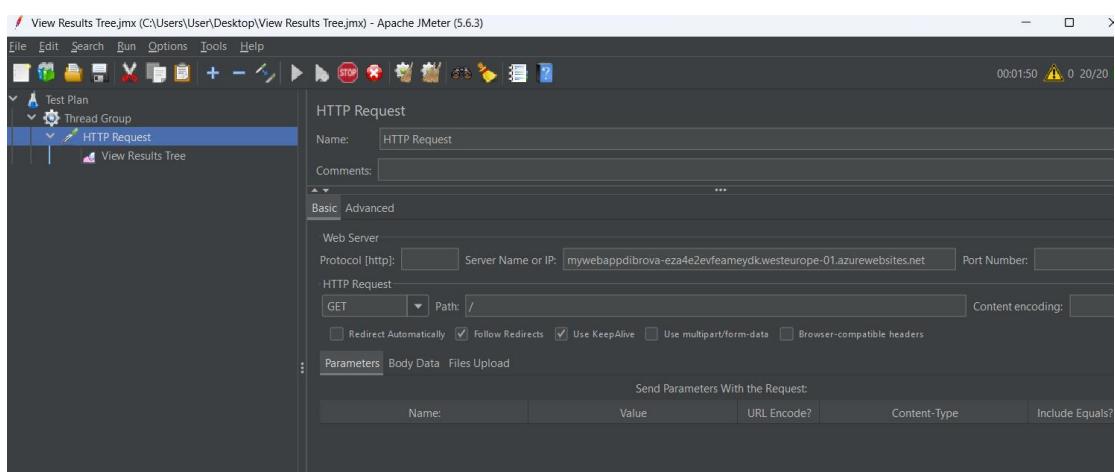
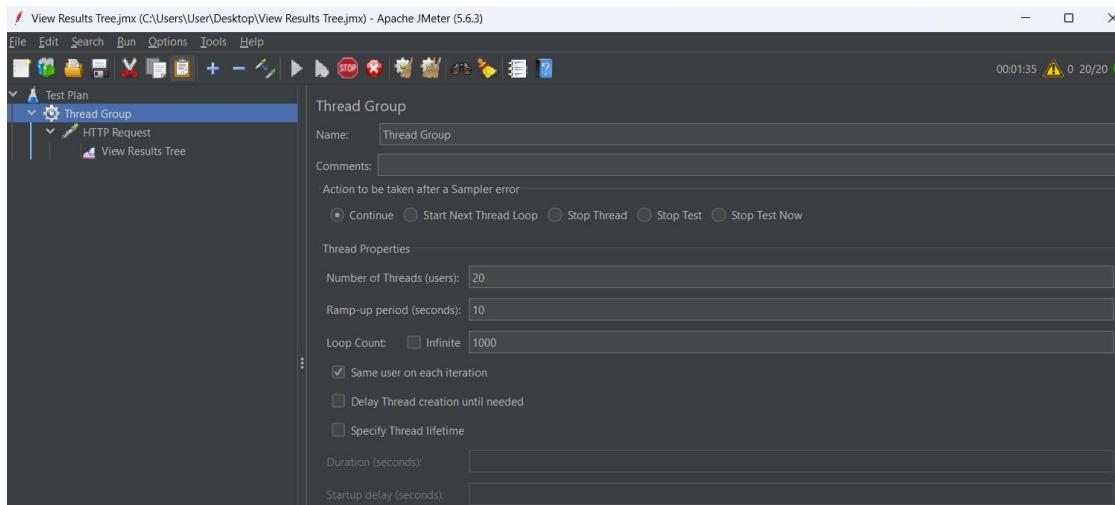
Overview Activity log Access control (IAM) Tags Diagnose and solve problems > Investigate < Monitoring Alerts Metrics Diagnostic settings < Logs Workbooks Usage Configure Properties Smart detection settings Network Isolation

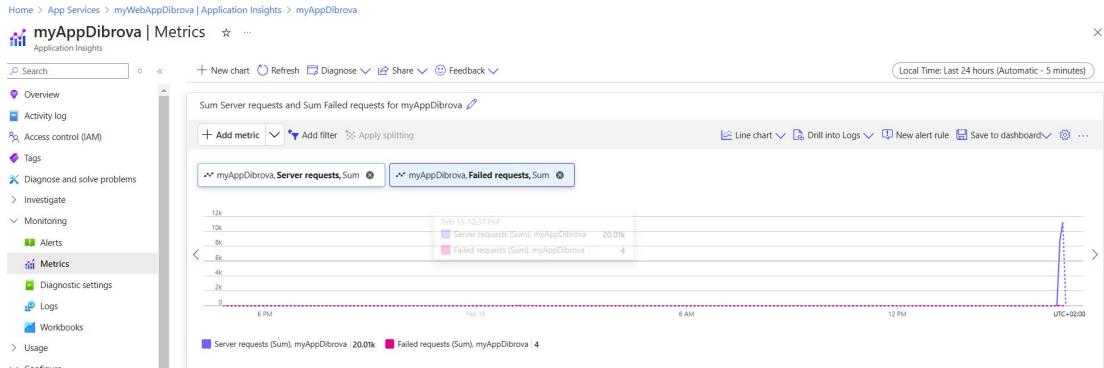


- **Server response**
- **Request count**
- **Failed request rate**

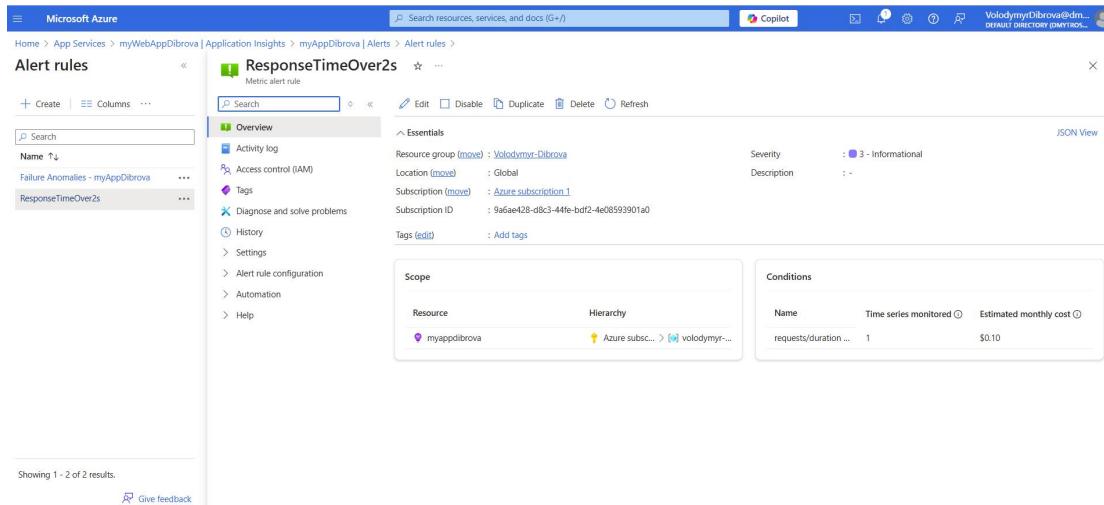
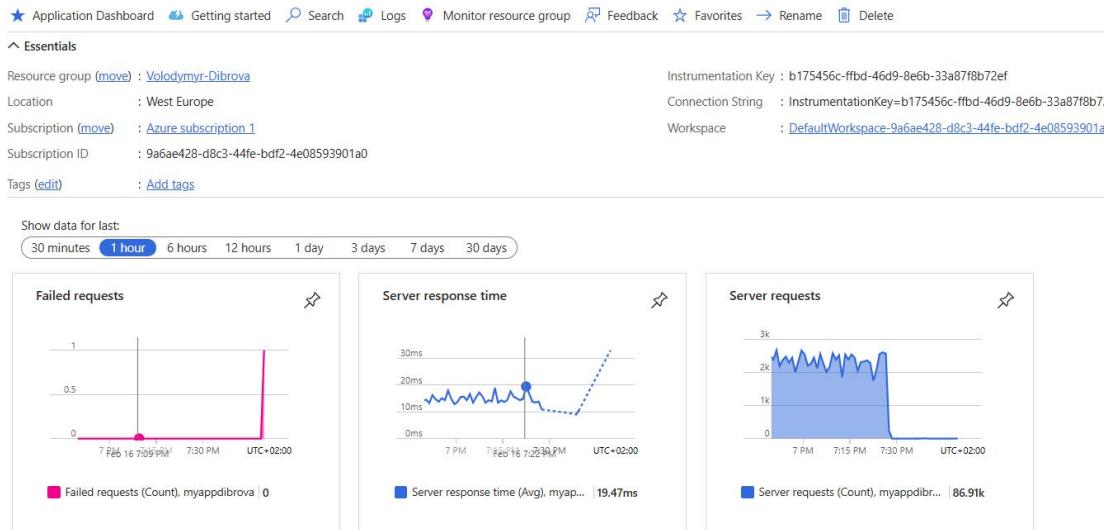
The screenshot shows the Microsoft Azure web app creation wizard. The current step is 'Your web app is running and waiting for your content'. It includes a note: 'Your web app is live, but we don't have your content yet. If you've already deployed, it could take up to 5 minutes for your content to show up, so come back soon.' To the right is an illustration of a computer monitor displaying a globe, with code snippets like '</>' and '{ }' floating around it. Below the note are two sections: 'Haven't deployed yet?' and 'Starting a new web site?'. A 'Quickstart' button is located between them. At the bottom are links for 'Deployment center' and 'Python'.

Generated application load using Jmeter





- **20,01k successful Server requests**
- **4 Failed requests**



Practical Task 16: Analyze Application Telemetry and Dependency Tracking Requirements:

Use Application Insights to:

- Enable **Live Metrics Stream** for real-time monitoring.
- Capture **custom events and telemetry** from an application.
- Track **dependencies** (database calls, external API calls).

- Write a simple **Kusto Query** to retrieve and analyze:
 - The slowest 10 requests in the last **24 hours**.
 - The most frequently failing requests.
 - Visualize the results in **Application Insights Workbooks**.

Configured for live metrics

Created Diagnostic settings

The screenshot shows the 'Diagnostic setting' configuration page. Under 'Logs', 'Metrics' and 'Dependencies' are selected. Under 'Destination details', 'Send to Log Analytics workspace' is checked, and the 'Subscription' is set to 'Azure subscription 1' with 'Log Analytics workspace' set to 'myLogVolodymyrDibrovaUS (eastus)'. Other options like 'Archive to a storage account', 'Stream to an event hub', and 'Send to partner solution' are also listed.

The screenshot shows the 'Live metrics' dashboard for 'myAppDibrova'. It displays real-time data for 'Outgoing Requests' (Dependency Call Rate, Dependency Call Duration, Dependency Call Failure Rate), 'Overall Health' (Committed Memory, CPU Total (%), Exception Rate), and 'Servers (10s avg)'. On the right, a 'Sample telemetry' section shows log entries with timestamps and exception details.

The screenshot shows the 'Logs' query editor for 'myAppDibrova'. A custom query is displayed:

```

1 // ...
2 // requestRateFriendlyName
3 let grainTime = 30min;
4 let grainPerSecond = 30.0 * 60.0; // Rate per second
5 requests
6 | where timestamp > ago(24h)
7 | summarize ["rate"] = sum(itemCount) / grainPerSecond by bin(timestamp, grainTime)
8 | render timechart
9
10 // requestDurationFriendlyName
11 requests

```

The results show a line chart of 'rate' over time, with values starting near zero, rising to a peak around 4 PM, dropping, and then rising again towards 6 PM.

Find the 10 slowest requests in the last 24 hours

Home > myAppDibrova

myAppDibrova | Logs

Application Insights

New Query 1*

myAppDibrova Select scope

Tables Queries Functions ...

Search Filter Group by: Solution

```

1 requests
2 | where timestamp > ago(24h)
3 | order by duration desc
4 | take 10
    
```

Run Time range: Set in query Save Share New alert rule Export Pin to Format query

Favorites

You can add favorites by clicking on the icon

Application Insights

- availabilityResults
- browserTimings
- customEvents
- customMetrics
- dependencies
- exceptions
- pageViews
- performanceCounters
- requests
- traces

Results

timestamp [UTC]	id	name	url	success	resultCode	duration	perform
2/16/2025, 3:32:45.823 PM	7ffd46633410bc	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	717	500ms
2/16/2025, 3:22:29.176 PM	edfb1f6e09bcb14e	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	695	500ms
2/16/2025, 3:37:49.430 PM	b59e3c3033760b	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	663	500ms
2/16/2025, 3:32:46.670 PM	982bd0dd9f512a2c	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	641	500ms
2/16/2025, 4:37:28.814 PM	4193eecd11306e	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	601	500ms
2/16/2025, 6:32:29.006 PM	8ff398b71cd8a04	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	584	500ms
2/16/2025, 6:32:29.006 PM	8ff398b71cd8a04	GET /	https://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	584	500ms
2/16/2025, 6:47:32.198 PM	30dadfb305fb1e1	GET /	http://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	583	500ms
2/16/2025, 6:47:32.198 PM	30dadfb305fb1e1	GET /	https://mywebappdibrova-eza4e2vefeameydk.westeurope...	True	200	583	500ms

Find the most frequent failed requests

myAppDibrova

myAppDibrova | Logs

Application Insights

New Query 1*

myAppDibrova Select scope

Tables Queries Functions ...

Search Filter Group by: Solution

```

1 requests
2 | where success == "False"
3 | summarize count() by name
4 | order by count_desc
    
```

Run Time range: Last 24 hours Save Share

Favorites

You can add favorites by clicking on the icon

Application Insights

- availabilityResults
- browserTimings

Results

name	count
GET /robots933456.txt	12

Check dependency failures

myAppDibrova

myAppDibrova | Logs

Application Insights

New Query 1*

myAppDibrova Select scope

Tables Queries Functions ...

Search Filter Group by: Solution

```

1 dependencies
2 | where success == "False"
3 | summarize count() by name
4 | order by count_desc
    
```

Run Time range: Last 24 hours Save Share

Favorites

You can add favorites by clicking on the icon

Application Insights

- availabilityResults
- browserTimings
- customEvents

Results

No results found from the last 24 hours
Try selecting another time range

Visualized the results in Application Insights Workbooks

The screenshot shows the Microsoft Azure Application Insights interface. On the left, there's a sidebar titled 'Application Insights' with a 'Default Directory' section containing links to 'kochubeiwebapp-insights', 'myAppDibrova', and 'mywebapp-1739707561'. The main area is titled 'myAppDibrova | Workbooks | Unsaved Workbook - 2/16/2025, 9:02 PM'. It displays a 'Workbooks' list and a 'Monitoring' section. A central window is titled 'Editing query item: query - 2' with tabs for 'Settings', 'Advanced Settings', 'Style', and 'Advanced Editor'. The 'Query' tab shows the Kusto Query Language (KQL) query:


```

    requests
    | where timestamp > ago(24h)
    | order by duration desc
    | take 10
    
```

 Below the query is a table with columns: timestamp, id, source, name, url, success, and resultCode. The table contains 10 rows of log data.

Log Analytics Tasks

Practical Task 17: Query and Analyze Azure Logs with Kusto Query Language (KQL)

Requirements:

- Connect **Azure Log Analytics** to a **Virtual Machine**.
- Use **Azure Monitor Logs** to ingest system logs.
- Write **basic KQL queries** to analyze logs:
 - Retrieve all logs from the last **3 hours**.
 - Find failed login attempts on the VM.
- Identify the top **processes consuming CPU resources**.
- Create a **scheduled query rule** to trigger an alert when a process exceeds **90% CPU usage**.
- Export the query results to a **CSV file** for reporting.

The screenshot shows the 'Diagnostic setting' configuration page for a Log Analytics workspace. At the top, there are buttons for 'Save', 'Discard', 'Delete', and 'Feedback'. Below that, a note states: 'A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)'.

Diagnostic setting name: VMLogsToLogAnalytics

Logs

Category groups	audit	allLogs
Categories	<input checked="" type="checkbox"/> Audit	<input type="checkbox"/> Summary Logs

Metrics

AllMetrics
<input type="checkbox"/>

Destination details

Send to Log Analytics workspace
<input checked="" type="checkbox"/>

Subscription: Azure subscription 1

Log Analytics workspace: myLogVolodymyrDibrovaUS (eastus)

Other options (unchecked):

- Archive to a storage account
- Stream to an event hub
- Send to partner solution

https://learn.microsoft.com/azure/azure-monitor/platform/diagnostic-settings?WT.mc_id=Port...

Search resources, services, and docs (G+)

Copilot

VolodymyrDibrova@dm...
DEFAULT DIRECTORY

New Query 1*

myLogVolodymyr... Select scope

Run Time range: Set in query Save Share New alert rule Export Pin to Format query

Tables Queries Functions ...

Search Filter Group by: Solution

Favorites

You can add favorites by clicking on the star icon

LogManagement

Results Chart

TimeGenerated [UTC] | SourceComputerId | ComputerIP | Computer | Category | OSType | OSName | OSMajorVersion

> 2/16/2025, 7:42:21.951 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:41:21.935 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:40:21.941 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:39:21.938 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:38:21.931 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:37:21.945 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:36:21.964 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:35:22.054 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:34:21.955 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18
> 2/16/2025, 7:33:21.941 PM	98efffb1-7526-db4c-90f0-ea3281bd1f80	172.178.50.233	myLinuxVM	Azure Monitor Agent	Linux	Ubuntu	18

0s 567ms | Display time (UTC+00:00) | Query details | 1 - 10 of 180

LogVolodymyrDibrovaUS | Logs

Analytics workspace

New Query 1*

myLogVolodymyr... Select scope

Run Time range: Last 24 hours Save Share New alert rule Export

Tables Queries Functions ...

Search Filter Group by: Solution

Favorites

You can add favorites by clicking on the star icon

LogManagement

Results Chart

TimeGenerated [UTC] ↑ | Computer | ObjectName | CounterName | InstanceName | CounterValue

> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	kcompactd0	0
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct Privileged Time	ib_nl_sa_wq	0
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct User Time	ib_nl_sa_wq	0
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Used Memory	ib_nl_sa_wq	0
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	ib_nl_sa_wq	0
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct Privileged Time	accounts-daemon	1.33
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct User Time	accounts-daemon	1.2
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Used Memory	accounts-daemon	5873664
> 2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	accounts-daemon	294912000

TOP 5 processes with the highest CPU usage in the last 3 hours.

myLogVolodymyrDibrovaUS | Logs

Log Analytics workspace

New Query 1*

myLogVolodymyr... Select scope

Run Time range: Set in query Save Share New alert rule Export

Tables Queries Functions ...

Search Filter Group by: Solution

Favorites

You can add favorites by clicking on the star icon

LogManagement

Results Chart

InstanceName | AvgCPUUsage

> cpu0	0.7637800125322971
> total	0.7537853507385099
> cpu1	0.7442263956506971

Created alert rule by Custom log search

Home > Monitor | Alerts >

Create an alert rule ...

Scope Condition Actions Details Tags Review + create

Configure when the alert rule should trigger by selecting a signal and defining its logic.

Signal name * ⓘ

Custom log search

[See all signals](#)

Define the logic for triggering an alert. Use the chart to view trends in the data. [Learn more](#)

The query to run on this resource's logs. The results returned by this query are used to populate the alert definition below.

Search query *

```
Perf
| where ObjectName == "Processor" and CounterName == "% Processor Time"
| where CounterValue > 80
| project TimeGenerated, Computer, InstanceName, CounterValue
```



[View result and edit query in Logs](#)

Measurement

Select how to summarize the results. We try to detect summarized data from the query results automatically.

Measure ⓘ

Table rows

[Review + create](#)

[Previous](#)

[Next: Actions >](#)

Exported the query results to a CSV file for reporting

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

VladimirDibrov/dm...
DEFAULT DIRECTORY (dmvty...)

Home > Monitor

Monitor | Logs

New Query 1*

MYLINUXVM Select scope

Run Time range: Last 24 hours Save Share + New alert rule Import Export Pin to Format query

Overview Activity log Alerts Metrics Logs Change Analysis Service health Workbooks Investigator (preview) Insights Managed Services Settings Diagnostic settings Data Collection Rules Data Collection Endpoints Azure Monitor pipelines (preview) Autoscale

Tables Queries Functions ...

Search Filter Group by: Resource type

Collapsible sidebar on the left.

Results Chart

TimeGenerated [UTC]	Computer	ObjectName	CounterName	InstanceIdName	CounterValue	CounterPath
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	kcompacted0	0	\myLinuxVM\Process\kcompacted0\Virtual Shared Memory
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct Privileged Time	ib_n_ls_wq	0	\myLinuxVM\Process\ib_n_ls_wq\Pct Privileged Time
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct User Time	ib_n_ls_wq	0	\myLinuxVM\Process\ib_n_ls_wq\Pct User Time
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Used Memory	ib_n_ls_wq	0	\myLinuxVM\Process\ib_n_ls_wq\Used Memory
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	accounts-daemon	0	\myLinuxVM\Process\accounts-daemon\Virtual Shared Memory
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct Privileged Time	accounts-daemon	1.33	\myLinuxVM\Process\accounts-daemon\Pct Privileged Time
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Pct User Time	accounts-daemon	1.2	\myLinuxVM\Process\accounts-daemon\Pct User Time
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Used Memory	accounts-daemon	5873664	\myLinuxVM\Process\accounts-daemon\Used Memory
2/16/2025, 7:24:00.000 PM	myLinuxVM	Process	Virtual Shared Memory	accounts-daemon	204812000	\myLinuxVM\Process\accounts-daemon\Virtual Shared Memory

0s 845ms Display time (UTC-0000) Query details 1 - 9 of 10